

POLITECNICO DI TORINO

Collegio di Ingegneria Informatica, del Cinema e Meccatronica

Corso di Laurea Magistrale

in Ingegneria del Cinema e dei Mezzi di Comunicazione

Tesi di Laurea Magistrale

**Studio del Workflow per l'uso di Motion Capture nella
Produzione Indipendente di un Prodotto di Animazione**



Relatore

prof. Antonino Riccardo Antonio Silvio

Candidato

Coarezza Melissa

Dicembre 2017

*“Devote yourself to an idea,
go make it happen, struggle on it, overcome your fears, smile,
don't you forget this is your dream.”*

Anonymous

I. Abstract

Lo scopo di questa tesi è stato lo studio delle principali tecnologie di Motion Capture al fine di valutare la migliore soluzione da adottare nella pipeline di produzione di una serie animata in CG. Tale studio si è svolto contestualmente alla produzione di un video in computer grafica che comprendesse l'utilizzo di una tuta provvista da sensori inerziali integrati per la registrazione del movimento. Il Motion Capture è una tecnica di animazione avanzata che ad oggi trova ampio margine di utilizzo non solo nell'industria videoludica, ma anche cinematografica e per installazioni museali in realtà virtuale e aumentata.

Il progetto si è diviso quindi in due fasi: la prima di ricerca, riguardante le tecnologie MoCap più diffuse ad oggi e quali disponibili per un mercato di produzione indipendente, e la seconda di produzione di un teaser trailer da utilizzare come presentazione della serie animata nei principali festival di animazione italiani e internazionali.

Nell'ambito del Motion Capture si è formata una partnership con l'azienda danese Rokoko, dalla quale è stata fornita a scopo di test la tuta inerziale Smartsuit Pro. In corso d'opera si è aperta la possibilità di beta testing di alcune funzioni del prodotto e delle possibilità di collaborazione con l'azienda.

La seconda parte della tesi si è svolta a Torino presso Robin Studio S.r.l.s ed ha previsto lo studio del workflow che ha portato alla produzione del teaser per il progetto di serie Namur. Il teaser è nello specifico un video interamente animato con tecniche mocap in ambiente Autodesk Maya, e ha seguito quindi tutte le fasi di sviluppo che si sono susseguite fino al raggiungimento del prodotto finito. In entrambi i momenti della ricerca ci sono stati momenti di riflessioni su alcune problematiche riscontrate, lo studio analitico del problema e sono state adottate diverse soluzioni.

Il teaser trailer prodotto sarà utilizzato per scopi promozionali e di crowdfunding del progetto, a livello nazionale e internazionale, poiché il progetto intende svilupparsi in una serie televisiva animata composta da più stagioni, dove la Motion Capture di tipo inerziale accompagnerà tutto il processo di animazione per raggiungere risultati di alto livello nell'ambito di una produzione di stampo indipendente.

II. Ringraziamenti

Il ringraziamento più importante è per i miei genitori, mi avete dimostrato come sia possibile vivere sempre al pieno delle mie forze e facoltà, mi avete insegnato che non esistono limiti e mi avete regalato delle esperienze meravigliose che altrimenti avrei solo potuto sognare. Los Angeles è stata l'esperienza di vita che mi ha insegnato a cavarmela da sola e a prendermi cura di me stessa e dei miei mali con un'attitudine positiva, Londra è solo la linea di arrivo delle prove che mi avete dato modo di superare per crescere la persona che sono oggi, e la professionista che sarò domani. Non so esprimere quanto io vi sia grata per questa vita meravigliosa e per tutte le opportunità che ne stanno nascendo, ma sappiate che vi adoro all'infinito, senza di voi non sarei neanche la metà della sognatrice che sono.

Grazie a mio fratello, che è il diretto responsabile (anche se non se ne rende conto, ma giuro che è così) delle mie aspirazioni di carriera, lui che da quando sono piccola è stato mio compagno di giochi e mi ha fatto conoscere il mondo della musica, ma soprattutto quello dei videogiochi: quando ti sei presentato a casa con Final Fantasy VIII hai firmato la nostra *condanna* e adesso grazie a te sto inseguendo un sogno bellissimo, così come te lo sei costruito tu, sin da quando eravamo piccoli. E poi perché non poteva esserci solo un ingegnere in casa, dovevo almeno diventare fortissima quanto te.

Ad Andrea, il mio immenso amore. Grazie per la pazienza e la tenacia che hai dimostrato in questi anni del mio essere uragano, posso solo immaginare quanto mi mancherai nei prossimi mesi e quanto sarai in pensiero, ma non vedo l'ora di tornare e iniziare a realizzare a tutti i nostri progetti di una vita insieme. Grazie di essere arrivato nel momento giusto, quando chimica sembrava uno scoglio insormontabile, e grazie di avermi spinto a realizzare tutti i miei sogni, se sto per partire per quest'avventura è solamente grazie a te che mi hai dato il coraggio di inseguire i miei progetti fino in fondo.

Un ringraziamento speciale al mio relatore, il prof. Antonino Riccardo, che ha creduto da subito nel mio progetto, e mi ha spronato a realizzarlo anche oltre gli obiettivi che mi ero prefissata. E' il docente di cui tutti gli studenti avrebbero bisogno, ed è stato un mentore che ha saputo vedere quali prendere le mie capacità e a trasformarle in una professione. Mi ha permesso di avere accesso a tutto ciò di cui avevo bisogno per completare questa tesi al meglio, spingendomi fino a Copenaghen e compiendo sacrifici che spero potrò ripagare molto presto.

A Drake, che sei cresciuto assistendomi nello studio e hai sempre trovato il momento giusto per costringermi a staccare dai libri per stare con te.

Indice dei Contenuti

| | |
|---|---|
| 1. Introduzione | 1 |
| 1.1 Motivazione | 2 |
| 1.2 Uno sguardo alle aziende italiane | 3 |
| 1.3 Outline..... | 4 |

Parte I - MOTION CAPTURE

| | |
|---|----|
| 2. Introduzione al Motion Capture | 7 |
| 2.1 Il problema dell'Uncanny Valley..... | 10 |
| 3. Sistemi di Motion Capture | 15 |
| 3.1 Sistema ottico..... | 15 |
| 3.2 Sistema meccanico..... | 17 |
| 3.3 Sistema elettromagnetico | 18 |
| 3.4 Sistema inerziale | 20 |
| 3.5 Tabella comparativa dei sistemi..... | 22 |
| 4. Motion Capture Facciale | 25 |
| 5. Pipeline per il Motion Capture | 29 |
| 5.1 Calibrazione | 30 |
| 5.2 Performance capture | 30 |
| 5.3 Elaborazione e pulizia dei dati | 31 |
| 5.4 Applicazione del movimento | 31 |
| 6. Case Study: Rokoko | 33 |
| 6.1 Smartsuit Pro e Smartuit Pro Studio | 34 |
| 6.2 Performance test..... | 36 |
| 6.3 Formulazione del problema | 41 |
| 6.4 Soluzione proposta e aggiornamento Smartsuit Studio | 44 |
| 7. Case Study: Faceware Technologies | 47 |

| | |
|---|----|
| 7.1 Faceware Analyzer..... | 48 |
| 7.2 Faceware Retargeting..... | 49 |
| 7.3 Pipeline per il Motion Capture Facciale | 51 |

Parte II - PRODUZIONE INDIPENDENTE

| | |
|--|------------|
| 8. Progetto Reverie Dawnfall..... | 53 |
| 8.1 Pre Produzione | 55 |
| 8.2 Personaggi..... | 56 |
| 8.2.1 Nadya Sinkamen | 57 |
| 8.2.2 Jameela Rani | 60 |
| 8.2.3 Peregrine | 61 |
| 8.2.4 Breather..... | 62 |
| 8.3 Ambientazione | 63 |
| 9. Teaser Trailer..... | 65 |
| 9.1 Sceneggiatura..... | 66 |
| 10. Pipeline di Produzione per il teaser..... | 67 |
| 10.1 Retopology..... | 68 |
| 10.2 Rigging..... | 70 |
| 10.2.1 HumanIK per il rigging del corpo..... | 73 |
| 10.2.2 Advanced Skeleton per il rigging facciale | 83 |
| 10.2.3 Dynamic Fk per il rigging dei capelli | 92 |
| 10.2.4 Rigging di Alep l'insetto | 94 |
| 10.2.5 Note sul rigging di Jameela..... | 95 |
| 10.3 Animazione | 96 |
| 10.3.1 Performance Capture | 97 |
| 10.3.2 Retargeting del corpo | 103 |
| 10.3.3 Retargeting del viso | 104 |
| 10.3.4 Animation Layers..... | 108 |
| 10.4 Rendering del prodotto finale | 111 |
| 11. Conclusioni..... | 117 |

| | |
|--|------------|
| Appendice A: Filtro di Eulero e Quaternioni per la prevenzione del Gimbal Lock..... | 121 |
| Bibliografia..... | 123 |
| Bibliografia classica..... | 123 |
| Sitografia..... | 123 |

1. Introduzione

Il motion capture è una tecnica di animazione avanzata che, sia nelle grandi produzioni cinematografiche che in quelle videoludiche, è molto frequente e richiesta per agevolare il compito degli animatori nell'impresa di rendere in maniera iperrealistica i personaggi digitali. E' diventata quasi un passaggio obbligatorio per le produzioni di alta qualità, ma non è detto che sia sempre stata vista di buon occhio dagli animatori più tradizionali. Uno dei vantaggi più acclamati del motion capture è sicuramente l'incremento della velocità della pipeline di produzione insieme a un alto grado di realismo, eppure gli animatori abituati a disegnare ogni singolo frame su carta hanno rappresentato uno scoglio per lo sviluppo iniziale di questa tecnologia a causa del loro scarso interesse nei dati prodotti dalle sessioni di performance capture. Dopo aver dovuto accettare l'ingresso dei computer nel loro spazio di lavoro, e digitalizzare così il processo di animazione, erano altresì ostinati nel riprodurre le animazioni a mano keyframe per keyframe e ogni tentativo di implementare il motion capture, sia del corpo o facciale, veniva percepito come un pericolo reale per la loro sicurezza nel mantenere un posto nell'industria.

Senza entrare nel dettaglio dell'animazione tradizionale, già trattata in letteratura in maniera più che esaustiva, questa ricerca vuole concentrare l'attenzione sul passo che sta intraprendendo, ormai da un paio di decenni, una fetta sempre maggiore dell'industria dell'animazione, e sugli sbocchi professionali che si possono creare sul suolo nazionale. Lo scopo di questa tesi è lo studio approfondito del motion capture e la sua applicazione durante la produzione di un contenuto d'animazione. Il MoCap sul nostro territorio è ancora considerato inaccessibile dati gli elevati costi di produzione o di acquisto: sono pochi gli studi di produzione, soprattutto nell'ambito dei videogiochi indipendenti, che stanno investendo nell'acquisto di queste nuove tecnologie, favorendo un costante sviluppo di alternative più economiche e quindi più facilmente accessibili. Fortunatamente, negli ultimi anni sono nati alcuni progetti di crowdfunding e casi startup di successo che hanno sviluppato degli strumenti di 'mocap' considerabili low cost: la *Smartsuit Pro* di Rokoko, per citarne una, costituisce una sezione fondamentale di questa ricerca. La produzione del contenuto in CG (Computer Graphics) è stato realizzato in un contesto totalmente indipendente, che ha inciso notevolmente sulle alcune delle scelte stilistiche.

1.1 Motivazione

Il moto di spinta verso questo argomento di tesi ha le sue radici in una passione radicata sin dall'infanzia, quella dei videogiochi. Il fascino di ogni singola cutscene all'interno di *Final Fantasy VIII* ha rappresentato l'inizio di un percorso che ad oggi si manifesta in un morboso interesse verso tutto quello che riguarda l'animazione cinematografica. In principio c'era la curiosità e il fascino vero il mondo del cinema e ancora di più degli effetti speciali, il poter ammirare luoghi fantasiosi e immensi capaci di affascinare oltre ogni immaginazione: per questo motivo ho intrapreso il percorso della Laurea triennale in Ingegneria del Cinema e dei Mezzi di Comunicazione al Politecnico di Torino. La svolta è avvenuta durante il corso di Computer Grafica, tenuto dal prof. Bottino: il mondo del 3D aveva spalancato le sue porte a noi laureandi, e da allora ha rappresentato una delle mie massime aspirazioni. I corsi magistrali di Effetti Speciali del prof. Antonino, nonché supervisore di questa ricerca, e di Computer Animation del prof. Sanna sono stati solo la chiusura di un cerchio e una forte conferma che il percorso che stavo intraprendendo aveva acquisito un senso. Rientrando nel merito dell'animazione cinematografica, ha iniziato a profilarsi il desiderio di scoprire e studiare nel dettaglio come questo tipo di lavoro si svolge, il suo attuale grado di complessità e gli strumenti di realizzazione.

Pertanto il lavoro di questa tesi si divide in due macro argomenti. Il primo riguarda l'analisi dello stato dell'arte del motion capture e il confronto tra diverse tecnologie, in particolare sarà studiata la tuta di Rokoko (*Smartsuit Pro*) e quali siano stati i limiti riscontrati durante i primi test con questo sistema. Per il motion capture facciale sarà presentato il pacchetto di programmi che serviranno per implementare nella pipeline anche il mocap per il viso. La seconda parte sarà dedicata allo sviluppo di una pipeline di produzione di una serie televisiva animata denominata attualmente *Reverie Dawnfall*, che raccoglie tutti gli studi effettuati nei primi mesi di ricerca.

L'aspetto chiave di questa ricerca è che ha offerto la possibilità di mettere in gioco una serie di conoscenze acquisite durante il percorso accademico e nello stesso tempo di evolvere a livello professionale. È importante far notare che la metodologia di ricerca e di sviluppo è stata costruita interamente da zero, andando ad acquisire in poco tempo diverse competenze nell'ambito dell'animazione, della computer grafica e della sua pipeline, e dei pacchetti software utilizzati come standard nell'industria. Lo svolgimento di questo lavoro di tesi è stato un susseguirsi di processi '*trial&error*' fino al raggiungimento dell'obiettivo desiderato,

di costante e rigoroso studio di tecniche totalmente nuove all'interno di un ambiente di lavoro in via di formazione, di stampo indipendente, e composto da personalità specializzate in settori della computer grafica differenti con i quali si è intrapreso un percorso di crescita comune e personale.

1.2 Uno sguardo alle aziende italiane

L'industria videoludica produce una quantità sempre maggiore di contenuti di altissima resa grafica per presentare un nuovo titolo: questi *cinematic* trailer vengono spesso commissionati a case di produzione esterne specializzate in questo tipo di prodotto digitale, che riescono a spingere al massimo le potenzialità della storia e dei personaggi che si vogliono rappresentare. In Italia l'industria videoludica così come quella dell'animazione e dei VFX si sta ancora sviluppando, con il risultato che le realtà più importanti si contano sulle dita di una mano e sono concentrate in Lombardia: si parla di Ubisoft e di Milestone per i videogiochi, e di EDI (Effetti Digitali Italiani) per gli effetti speciali. Vale la pena menzionare la Maga Animation Studio, radicata a Monza, che ha recentemente collaborato con Ubisoft Milan al recente lancio del titolo "*Mario + Rabbids Kingdom Battle*", di cui tra le altre mansioni ha curato le cinematiche del gioco. La divisione italiana di Ubisoft sta espandendo il suo organico proprio in questi mesi, anche grazie al successo del titolo appena citato, non tanto in merito ai numeri di vendita quanto alla visibilità che ha portato gli sviluppatori e i creativi dello studio italiano in questione.

Forge Reply sta investendo molto nella realtà virtuale e aumentata in ambito edutainment e propriamente ludico, e insieme al gruppo Reply di cui fa parte è da considerarsi un'altra grande realtà presente nel panorama nazionale. Di fatto, il quadro dell'industria videoludica negli ultimi anni è ampiamente positivo, come attestano i rapporti annuali dell'AESVI (Associazione Editori Sviluppatori Videogiochi Italiani) e se da un lato questo dato incoraggia la sua crescita, lo stesso non si può dire dell'industria degli effetti speciali: la computer grafica trova ampio utilizzo negli studi pubblicitari e piccole imprese di animazione, ma molto spesso localizzate all'interno dei confini del nostro Paese. Manca un polo per i VFX e l'animazione capace di competere con le grandi industrie mondiali, come MPC, Framestore e Sony Pictures per citarne alcune.

La fase sperimentale di questa tesi si pone come una sfida al mondo dell'animazione, con l'intenzione di produrre come studio indipendente un prodotto seriale utilizzando le nuove e più recenti tecnologie, come appunto il motion capture.

1.3 Outline

I capitoli di questa tesi sono suddivisi come segue:

Capitolo 2: Si argomenta la storia del Motion Capture nei suoi punti chiave, con particolare attenzione alla teoria dell'*Uncanny Valley* e un'analisi del suo utilizzo nell'industria dell'intrattenimento.

Capitolo 3-4: Si elencano i sistemi di Motion Capture disponibili ad oggi sul mercato, evidenziandone punti di forza e difetti. Si conclude la trattazione con una tabella comparativa dei vari sistemi e si riprende con una descrizione sul motion capture facciale e le differenze di impostazione rispetto a quello full-body.

Capitolo 5: Si presenta la composizione di una generica pipeline del Motion Capture, così come studiata per l'animazione del teaser trailer di *Reverie Dawnfall*.

Capitolo 6: Un approfondimento della tuta di MoCap inerziale Smartsuit Pro di Rokoko, che è stata fornita per il lavoro di questa ricerca.

Capitolo 7: Un approfondimento sugli strumenti software di Faceware Technologies per il motion capture, per il quale è stata ottenuta una licenza da utilizzare durante lo sviluppo del trailer.

Capitolo 8-9: Si presenta il progetto *Reverie Dawnfall*, una serie d'animazione che verrà prodotta presso Robin Studio S.r.l.s. Verranno presentati brevemente i personaggi, l'ambientazione e il contenuto del teaser trailer.

Capitolo 10: Viene trattato nel dettaglio il flusso di lavoro applicato alla produzione che ha portato alla realizzazione del video animato in CGI, in particolare verrà dato un resoconto del lavoro di rigging e di animazione che è stato personalmente compiuto come oggetto pratico di questo lavoro di tesi, con l'utilizzo delle tecnologie applicate a tecniche più tradizionali.

Capitolo 11: Una breve riflessione sui risultati raggiunti grazie alla tesi e una panoramica degli sviluppi futuri del progetto *Reverie Dawnfall*.

Parte I

Motion Capture

Lo studio presentato in questi capitoli si è sempre svolto nell’ottica di approfondire la conoscenza personale di tool di animazione avanzati da utilizzare durante la pipeline di animazione. Il Motion Capture costituisce uno degli elementi principali in un prodotto animato di qualità e massimo realismo, e si è voluta studiare la tecnologia sotto ogni aspetto, per analizzare quali siano le soluzioni più appropriate al contesto di lavoro nel quale si svolgerà la fase sperimentale di questa ricerca. All’interno di questa sezione si darà ampio spazio al confronto dei sistemi di motion capture per il corpo e per il viso, e si presenteranno due casi studio che hanno offerto il primo approccio sperimentale: la visita alla compagnia Rokoko a Copenaghen, fornitore della tuta inerziale Smartsuit Pro, e l’utilizzo licenziato ufficialmente da Faceware Technologies per testare il motion capture facciale con un software leader sul mercato.

2. Introduzione al Motion Capture

La tecnologia del Motion Capture (MoCap) nasce per soddisfare l’esigenza di voler campionare e rappresentare numericamente i movimenti di soggetti umani, animali o inanimati. Non a caso le prime aree di utilizzo di questo processo sono da attribuire al settore clinico o militare, dove la registrazione dei movimenti del soggetto in esame permettevano di valutarne eventuali problemi di postura o di analizzare la prestazione fisica. Per dare una definizione più generale si può dire che l’utilizzo del mocap è strettamente legato allo studio della cinematica di oggetti in movimento all’interno di un volume, e i parametri principali che vengono coinvolti sono lo spostamento, la velocità e l’accelerazione. I dati generati dal motion capture possono essere studiati e rielaborati a seconda del caso di utilizzo, in ambito medico ad esempio si può valutare l’abilità motoria del paziente e poter intervenire in maniera mirata con terapie apposite.

La nascita di veri e propri sistemi di motion capture avviene dall’inizio degli anni Ottanta, negli ambienti universitari e di ricerca, dove lo studio del movimento favorì lo

sviluppo di sistemi utili allo scopo. Nel 1982 al MIT viene presentata la *Graphical Marionette*, un sistema ottico che prevedeva l'uso di una serie di led posizionati su una tuta in corrispondenza dei giunti delle articolazioni, e quello di un paio di camere in grado di registrare le informazioni di movimento, con la possibilità di visualizzare in real-time appunto su una *marionetta* digitale.

Pochi anni dopo, nel 1988, Silicon Graphics e Pacific Data Images (PDI) presentarono al pubblico *Waldo*, un sistema che permetteva di gestire in real-time i movimenti della bocca di un personaggio a bassa risoluzione: in un video della rubrica *Jim Henson Hour*, viene spiegato che “*per una scena di due minuti di Waldo, al computer servono 120 ore per creare l'immagine finale ad alta risoluzione*”.

Nel 1995 fu pubblicato *FX Fighter*, il primo videogioco picchiaduro realizzato in 3D (personaggi e ambienti), e la peculiarità di questo titolo fu quella di registrare i set di mosse dei personaggi tramite motion capture, azionati dai tasti del giocatore in modo da non notare le transizioni intermedie: il successo di questo titolo incoraggiò altre case di produzione a investire nella nuova tecnologia.



Figura 2.1: Jim Henson e il sistema Waldo C. Graphic. A destra una rappresentazione a bassa risoluzione di Waldo (cliccare sull'immagine per andare al video)

La metodologia più seguita e più affidabile rimane ad oggi quella dei sistemi *marker based*, che verranno trattati nello specifico, spesso oggetti sferici di piccole dimensioni appiccicati in posizioni strategiche delle articolazioni su una tuta indossata dall'attore, che possono emettere o riflettere la luce per l'acquisizione del movimento. I dati vengono

processati dal calcolatore che fornisce una curva continua del movimento, ma non sempre è stato necessario l'utilizzo del computer per poter fare del motion capture.

Storicamente il motion capture nell'industria dell'animazione è associato alla tecnica del rotoscoping, sviluppata nel 1914 da Max Fleischer, ovvero un processo che permetteva agli animatori di ricalcare le pose a partire da immagini registrate di attori reali che venivano proiettate su un pannello di vetro traslucido, strumento di supporto per l'attività di disegno nell'animazione tradizionale: il prodotto più celebre di Fleischer a dimostrazione di questa tecnica è la serie animata *Out of the Inkwell* dove i suoi personaggi apparivano molto fluidi nei movimenti. Nel 1921 fondò insieme a suo fratello i Fleischer Studios, che diedero vita a un personaggio importante per i cartoni animati di quegli anni, Betty Boop (1931), tanto da diventare i diretti concorrenti di Disney perché i loro personaggi offrivano un umorismo e una sensualità adatta a un pubblico più maturo. Disney d'altra parte, si mostrò subito interessata al rotoscoping e produsse come primo film d'animazione a usare questa tecnica *Biancaneve e i Sette Nani*, che debuttò sul grande schermo nel 1937. Il lavoro degli animatori tradizionali ha sempre attinto a piene mani dalle immagini reali, e il rotoscoping permise di raggiungere un nuovo grado di naturalezza nei movimenti dei personaggi animati frame per frame. Dopo il successo di *Biancaneve*, il rotoscoping è stato usato da Disney per siglare altri successi come *La Sirenetta* (1989). Usare del materiale reale come reference per gli animatori divenne una prassi per i grandi classici Disney, e per altre case di produzione che ne seguirono l'esempio.



Figura 2.2: Esempio di rotoscoping usato in 'Biancaneve e i Sette Nani'

Nel frattempo, nel mondo del cinema e dell'intrattenimento nasceva una sempre più impellente richiesta di realismo nei personaggi digitali rappresentati sullo schermo, e seguendo lo schema e gli studi dell'*Uncanny Valley* si può ripercorrere l'ascesa di questa

tecnologia nei suoi successi e fallimenti al botteghino, che non sempre riflettono lo stesso grado di insoddisfazione nell'evoluzione della tecnologia ma piuttosto ne favorirono la divulgazione e la successiva sperimentazione per il miglioramento sempre costante del metodo.

2.1 Il problema dell'Uncanny Valley

Traducibile come *'la zona perturbante'*, l'Uncanny Valley è uno studio degli anni '70 nell'ambito della robotica, secondo cui la visione di replicanti e automi antropomorfi generi in un campione di persone un senso di familiarità e agio tanto più questi sono somiglianti alla figura umana. Nel grafico mostrato in Figura 2.3 però si può notare come ci sia un brusco calo e un aumento di reazione di disagio di fronte a un realismo considerato estremo o un commento del tipo *'ci sei quasi ma ancora no'*. Questo senso di sgradevolezza si vede riflesso in film d'animazione come *Polar Express* (2004) di Robert Zemeckis e *Final Fantasy: The Spirit Within* (2001) di Hironobu Sakaguchi dove le critiche più aspre furono mosse nei confronti degli occhi inespressivi e delle facce troppo plastiche. Nonostante *Polar Express* fu accolto positivamente nel suo complesso, lo stesso non si può dire dei successivi esperimenti di Zemeckis nello stesso ambito: *La Leggenda di Beowulf* (2007) e *Christmas Carol* (2009) hanno portato a un abbandono del regista di questa tecnica. Lo stesso vale per i dati d'incasso dei successivi film del franchise nipponico, *Final Fantasy VII: Advent Children* (2005) e *Kingsglaive: Final Fantasy XV* (2016), tralasciando il fatto che rimangono prodotti riservati allo scenario videoludico e quindi di base presentano dei numeri nettamente in svantaggio rispetto ai prodotti puramente cinematografici. Un altro esempio è il lungometraggio animato *Sinbad: Beyond the Veil of Mysts* (2000) accreditato come il primo film del suo genere ad essere girato esclusivamente in motion capture, prendendo come attori di riferimento figure del calibro di Mark Hamill, Brendan Fraser e John Rhys-Davies.

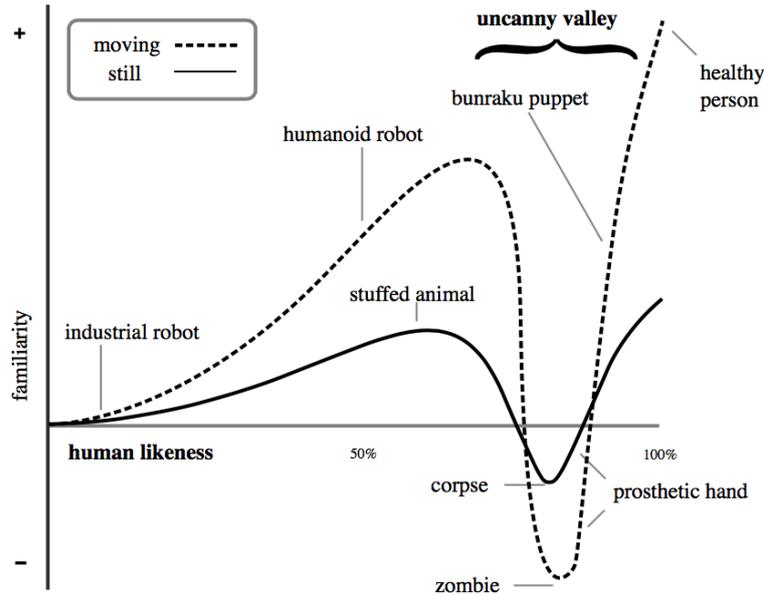


Figura 2.3: *Uncanny Valley*, grafico della reazione emotiva

Se il botteghino non registrò degli indiscutibili successi, non si può dire lo stesso a livello tecnico poiché la percezione di straniamento da parte dello spettatore spesso non coincide con il giudizio oggettivo che gli addetti ai lavori riservano riguardo alla resa visiva. Come per tutte le nuove tecnologie, ci vuole del tempo perché il pubblico sia in grado di apprezzarne le potenzialità in toto, ma non per questo l'industria deve investire di meno. Tutti i titoli appena citati si rivelarono un successo acclamato per la qualità grafica e le prestazioni ottenute dalla performance capture, registrando miglioramenti a livello di tecnologia che si ripercossero in investimenti sempre maggiori verso il motion capture, e che si riversarono in prodotti videoludici o in pellicola di qualità sempre maggiore, come nel caso di *Kingsglaiive: FFXV*.

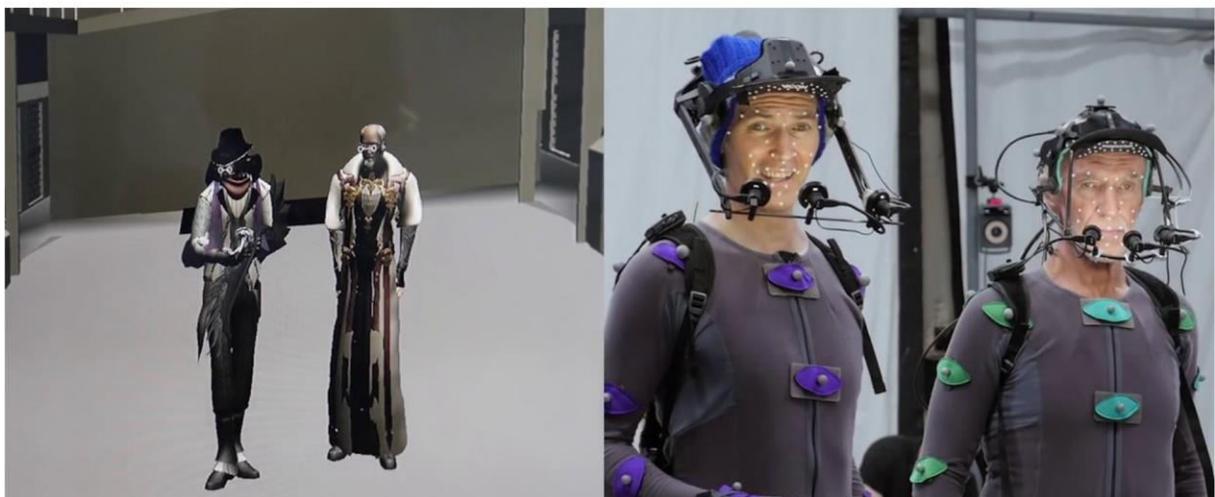


Figura 2.4: Motion capture per il film in CGI *'Kingsglaive: Final Fantasy XV'* (cliccare sull'immagine per andare al video)

Fino adesso sono stati presi in considerazione solamente film realizzati interamente in computer grafica per aprire una parentesi sul panorama del motion capture in questo ambito, che è lo stesso per il quale si svolge il progetto *Reverie Dawnfall*, ma è chiaro che il mocap si è sviluppato parallelamente nell'industria cinematografica, riscuotendo forse ancora più successo. Grazie ai grandi colossal come *Lo Hobbit: La Desolazione di Smaug* (2014) dove Benedict Cumberbatch ha impersonato un enorme drago regalando una prova attoriale capace di mettere in ombra addirittura il pioniere della performance capture Andy Serkis e il suo Gollum, il motion capture nel cinema gode di ottima salute e ormai globalmente accettato dal pubblico. Si tende a notare come il fenomeno dell'Uncanny Valley viene meno di fronte a personaggi non-umanoidi per cui lo spettatore non immagina l'impersonificazione umana dietro il lavoro che è stato svolto per ottenere un risultato credibile: oltre al drago Smaug è possibile citare Cesare e Koba, le scimmie protagoniste del prequel di *Planets of Apes*, impersonificate rispettivamente dal celebre Andy Serkis e da Toby Kebbel.

Una nota di merito va a James Cameron che durante le riprese del suo film di maggior successo, *Avatar* (2009), introdusse il concetto di *Virtual Camera* ovvero la possibilità di osservare in tempo reale il prodotto della performance capture dell'attore reale sulla controparte digitale: per il risultato ottenuto nel motion e per la novità portata anche dall'introduzione del 3D al cinema (di cui *Avatar* fu il primo esperimento di successo che di fatto introdusse quest'altra tecnologia nelle sale cinematografiche), *Avatar* è ad oggi il film in testa alle classifiche mondiali di incassi con un totale di \$2,783,918,982.

3. Sistemi di Motion Capture

I sistemi di motion capture si possono dividere sostanzialmente in 3 gruppi principali: sistemi ottici, sistemi magnetici e sistemi meccanici. A questi si aggiungono ulteriori categorie nate più di recente come i sistemi markerless e i sistemi acustici, mentre per quanto riguarda il mocap facciale si rimanda al capitolo successivo dedicato, pur ricordando che la strumentazione può essere integrata ai sistemi appena citati per poter lavorare su una registrazione di tipo full-body. Di seguito si presentano in breve le caratteristiche di ogni strumentazione, corredata infine con una tabella di comparazione tra le varie tipologie. Un paragrafo sarà dedicato ai sistemi inerziali, oggetto in uso per la fase sperimentale di questa ricerca e di ampio utilizzo nelle produzioni indipendenti.

3.1 Sistema ottico

Sicuramente il più diffuso a livello di tecnologia nell'industria cinematografica e delle grandi produzioni videoludiche, questo sistema marker-based utilizza un sistema di telecamere che si interfacciano con i marker posizionati sulla tuta in prossimità dei giunti sulla tuta. La dimensione e la forma dei marker dipende dalla risoluzione della camera, e possono essere di due tipologie:

- **Marker passivi**, sono composti da materiale retroriflettente di forma sferica che riflettono la luce che deriva da dei diodi luminescenti (LEDs) equipaggiati dalla camera, che ha una frequenza di acquisizione di 5000fps.
- **Marker attivi** - in questo caso i marker sono propriamente dei LEDs ad infrarossi che inviano dei segnali luminosi alle camere, possono illuminarsi singolarmente evitando così il bisogno di identificare ogni marker; altri sistemi marker-based attivi si illuminano in contemporanea e serve controllare l'ampiezza e la frequenza del segnale per distinguere i diversi marker.

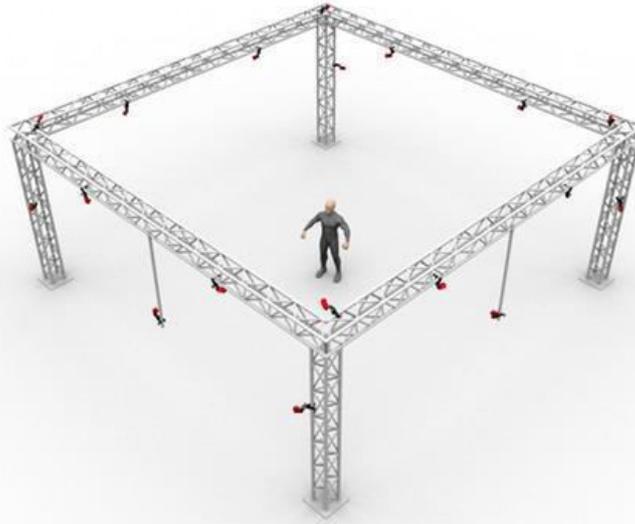


Figura 3.1: Largo volume di cattura per un sistema di motion capture ottico.

Per poter triangolare un marker nello spazio 3D sono necessarie almeno due camere ma è consigliato l'utilizzo di almeno tre camere per avere una maggiore precisione. I sistemi più professionali sono dotati di ampi ambienti che arrivano a supportare fino a 48 camere, per poter registrare più attori contemporaneamente e per limitare il più possibile le occlusioni. I vantaggi che offre questo tipo di sistema sono molteplici:

- I dati ottici sono molto accurati e il rate di registrazione è molto alto (30~2000 samples/s);
- Si possono registrare più soggetti in contemporanea e quindi utilizzare un ampio numero di markers;
- Lo spazio di cattura è uno dei più grandi rispetto ad altri sistemi, ciò comporta un'elevata libertà di movimento;
- I settaggi dei markers sono facili da variare a seconda del progetto;
- Si possono generare dati skeletal.

Gli svantaggi dei sistemi ottici invece riguardano più la parte di post-processing dei dati generati dal sistema, in particolare si riscontrano le seguenti problematiche:

- I dati rotazionali devono essere computati dai quelli traslazionali in post-processing poiché i dati prodotti sono limitati a 3 DOF (degrees of freedom);
- Nonostante si possa limitare, l'occlusione dei markers è sempre probabile;

- Inizialmente i dati in real-time sono visualizzabili solamente su modelli semplici (stick&bones);
- Elevati costi di hardware, i più alti rispetto agli altri sistemi.

Leader nel settore del motion capture ottico è Vicon, presente sul mercato dai primi anni Ottanta e tuttora considerata lo standard nelle produzioni di alto livello. L'evoluzione del loro sistema rende possibile visualizzare in real-time la performance dell'attore applicata al personaggio in CGI: l'esempio più recente e più spettacolare di questa applicazione è stata mostrata in alcune interviste fatte allo studio Ninja Theory durante lo sviluppo del titolo *Hellblade: Senua's Sacrifice* (2017) dove durante le registrazioni potevano visualizzare come sarebbe stato finalizzato il prodotto, su uno schermo posto nell'ambiente.



Figura 3.2: Registrazione motion capture e visualizzazione real-time all'interno del motore di gioco.

3.2 Sistema meccanico

I sistemi meccanici presentano le sembianze di un esoscheletro composto da una struttura di metallo o plastica rigida e potenziometri posizionati lungo i giunti del corpo, in grado di muoversi conformemente con l'attore. La potenzialità di questo tipo di sistema risiede nel fatto che sono markerless, lavorano in real-time e relativamente economici. Il rovescio negativo della medaglia è che la configurazione è piuttosto predefinita e soffre della mancanza di una certa libertà di settaggio dei parametri, senza contare che essendoci solo potenziometri è impossibile per il sistema definire la posizione globale dell'attore: compiendo un salto o salendo le scale, il sistema non mostrerà tali movimenti e nell'ultimo caso si avrà la sensazione di camminare sul posto. Per cercare di risolvere il problema si possono aggiungere dei sensori magnetici. In maniera schematica si riportano i punti di forza di questo sistema:

- Relativamente economico rispetto ad altri sistemi;
- Lavora in real-time in ampi volumi di cattura;
- Non ci sono occlusioni perchè non sono necessarie camere per la registrazione dei movimenti, né interferenze magnetiche o elettriche.

Gli svantaggi riguardano essenzialmente la fragilità del sistema esoscheletrico e alcune problematiche precedentemente citate:

- Non c'è una misurazione della traslazione globale;
- L'esoscheletro limita i movimenti dell'attore;
- Fragile nella sua composizione;
- Basso rate di registrazione e configurazione dei sensori pre-determinata.



Figura 3.3: Primi esperimenti di sistemi meccanici (sinistra) e sistemi di mocap facciale (destra)

3.3 Sistema elettromagnetico

Consistono in una serie di sensori, tipicamente tra la dozzina e la ventina, che posti sull'attore misurano la relazione spaziale rispetto a un emettitore di campo magnetico. Il primissimo vantaggio di questo sistema è la possibilità di utilizzo per applicazioni in tempo reale in quanto i sensori computano entrambe le componenti, rotazionale e traslazionale, quindi non richiedono ulteriori calcoli in fase di processing del compilatore. I sensori sono tipicamente a 6DOF e si possono ottenere dei risultati molto precisi se si accostano ai sistemi ottici che richiedono un numero più ampio di trackers; stesso discorso vale per il volume di cattura, non

si è vincolati a uno spazio chiuso sotto lo sguardo vigile di camere, ma è essenziale non lavorare con questi sistemi in spazi dove sono presenti materiali ad alta conduttività perché sono molto sensibili alle interferenze magnetiche.

Si possono distinguere due diverse categorie di questi sistemi, in base all'utilizzo della corrente:

- 1) AC (corrente alternata), lavorano con impulsi di onde quadre e sono molto sensibili al rame e all'alluminio;
- 2) DC (corrente continua), in questo caso gli impulsi sono rappresentati da un'onda sinusoidale e sono molto sensibili al ferro e all'acciaio.

I vantaggi sono molteplici se messi a confronto con i più usuali sistemi ottici:

- Posizione e rotazione sono automaticamente generati dal sistema;
- Molto più economici rispetto ai sistemi ottici;
- Adatti per applicazioni real-time, anche per più soggetti in contemporanea;
- Occlusioni da parte di oggetti magnetici assenti.

D'altra parte, il fatto di lavorare con delle componenti magnetiche rende questi sistemi difficili da gestire in ambienti non perfettamente adibiti al loro utilizzo:

- Subiscono sia interferenze magnetiche che elettriche e i volumi di cattura sono ridotti rispetto agli altri sistemi;
- Il loro utilizzo è vincolato alla durata delle batterie;
- Minor sampling rate rispetto ai sistemi ottici (144-200 samples/s) e dati rumorosi;
- Come per i sistemi meccanici, la configurazione dei sensori è limitata.



Figura 3.4: Sistema magnetico *Nest of Birds* di Ascension Technology (sinistra), tuta di motion capture magnetica (destra)

3.4 Sistema inerziale

Stanno acquisendo una maggiore fetta di mercato quei sistemi composti da sensori inerziali miniaturizzati, composti dalla fusione di tre diverse tipologie di sensori: sono sempre presenti un **giroscopio** in grado di misurare le rotazioni del dispositivo cui è applicato, e un **accelerometro** che calcola l'inerzia della massa sottoposta ad accelerazione per poter determinare inclinazione e posizione dell'apparecchio (entrambi i sensori possono essere tri-assiali, o in alcune versioni può esserci un sensore per ogni asse, indipendente dagli altri). Questi due elementi rendono il sistema a 6 gradi di libertà che può espandersi a 9 DOF in presenza di un **magnetometro** (anche questo tri-assiale o tre sensori separati), cioè un sensore che rileva l'intensità del campo magnetico circostante. L'insieme di questi sensori genera un sistema denominato IMU (Inertial Measurement Unit), che è più tecnicamente detto Sistema Micro-Elettro-Meccanico (MEMS).

L'ortogonalità degli assi è molto importante per i calcoli delle matrici di orientamento: in un accelerometro tri-assiale ad esempio, il sensore genera tre segnali analogici che descrivono l'accelerazione prodotta lungo ciascuno degli assi e agiscono sull'oggetto a cui è collegato. La più significativa di queste accelerazioni sensibili è l'accelerazione causata dalla gravità. Esistono principalmente due tipi di giroscopio:

- 1) **giroscopio ottico**, o integrante, misura i cambiamenti nell'orientamento;
- 2) **giroscopio di velocità** misura le variazioni della velocità angolare.

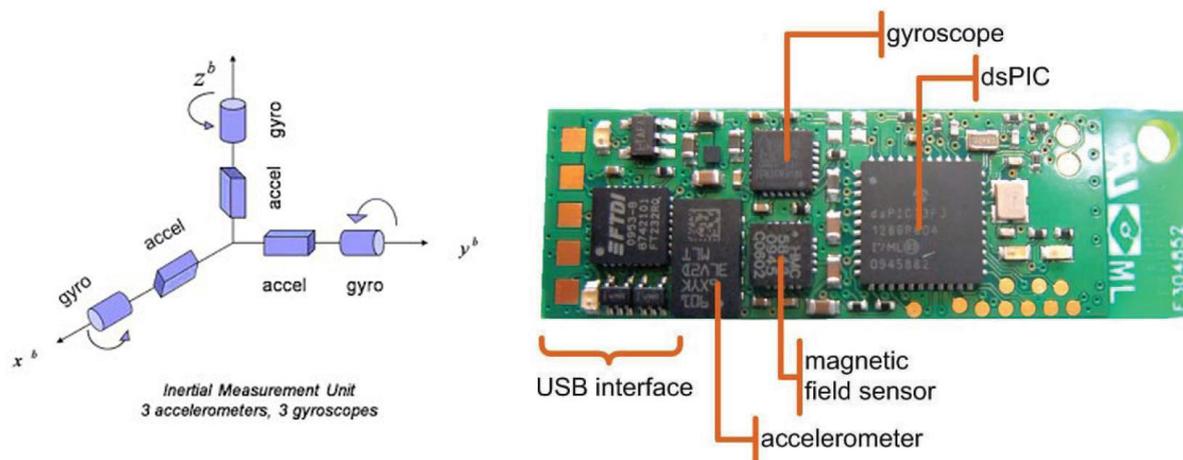


Figura 3.5: Accelerometri e giroscopi tri-asse impostati ortogonalmente, 6DOF (sinistra), esempio di sensore IMU a 9DOF (destra)

I magnetometri vengono inseriti nelle IMU più moderne per poter stabilire l'orientamento iniziale della posizione del soggetto nello spazio, e tutte le misurazioni successive vengono derivate integrando i dati dei giroscopi ottici.

Nei sistemi odierni più diffusi, come quelli offerti da XSens, Rokoko e Noitom, i sensori sono posti lungo segmenti delle articolazioni tramite supporti di tessuto come tute aderenti (XSens, Rokoko) o velcro adesivo (Noitom): in generale un hub centrale raccoglie le informazioni di tutti i sensori legati tra di loro con dei cavi, e invia in wireless le informazioni al computer che può visualizzare in real-time anche più soggetti contemporaneamente, a seconda delle limitazioni del programma.

Come nei sistemi meccanici, una forte limitazione di questi sistemi è data dalla mancanza di una misurazione globale per la componente traslazionale, che si ripercuote in errori di misurazione nel momento in cui si varia la posizione del soggetto sull'asse dell'altezza (es. salto o salita di un gradino). I vantaggi dei sistemi inerziali sono tra i più apprezzati dagli studi indipendenti:

- Nessuna occlusione e spazio di cattura potenzialmente illimitato;
- Visualizzazione in real-time senza ricorrere a post-processing;
- Cattura di molteplici soggetti;
- Molto più economici rispetto ai sistemi ottici e magnetici.

Tra i punti a sfavore del mocap inerziale si ricordano:

2. Non è possibile calcolare la posizione globale, è necessario ricorrere ad algoritmi di sensor fusion, filtri di Kalman avanzati oppure creare un sistema ibrido-ottico;
3. I sensori IMU possono soffrire di drifting ovvero errore nella misurazione della posizione nel tempo, per questo si predilige l'accostamento di magnetometri;
4. Lo spazio di cattura è *limitato* dal range della connessione wireless, è consigliato l'utilizzo di hotspot dedicati, le cui caratteristiche dipendono dalla scheda tecnica del prodotto.



Figura 3.6: Perception Neuron di Noiom (sinistra), XSens' MVN (centro), Smartuist Pro di Rokoko (destra)

Nell'ambito di questa ricerca è stata acquisita la tuta di motion capture inerziale Smartsuit Pro di Rokoko, ed è stato possibile sperimentare e valutare di persona i limiti del sistema. Ricontrando le problematiche già note in letteratura, si è voluto studiare diverse soluzioni teoriche per il calcolo della componente globale, in quanto all'avanzare della produzione animata si richiede la necessità di poter performare azioni concitate come momenti di parkour, o azioni più comuni come la possibilità di registrare correttamente un salto o uno sbalzo di altezza. Nel capitolo successivo dedicato all'analisi della Smartsuit Pro si analizzerà un po' più nel dettaglio la formulazione del problema e tramite documentazione da fonti reperibili in letteratura verranno analizzate alcune delle soluzioni implementabili, con un commento sulla migliore metodologia che si intenderà utilizzare nell'avanzamento del progetto.

3.5 Tabella comparativa dei sistemi

Di seguito si vuole riportare uno schema riassuntivo e comparativo tra i vari punti chiave che sono stati citati nei paragrafi precedenti, per poter tracciare un confronto globale tra i diversi modelli presenti sul mercato.

| Sistema | Tipologia | DOF | Occlusioni | Real-time | Sampling rate (Hz) | Costo |
|------------------|--------------|-----|------------|-----------|--------------------|---------------|
| Ottico | marker-based | 3 | ✓ | ✓ | high | \$\$\$\$\$ |
| Meccanico | markerless | 6 | - | ✓ | medium | \$\$ ~ \$\$\$ |
| Elettromagnetico | markerless | 6 | - | ✓ | low | \$\$ |
| Inerziale | markerless | 6-9 | - | ✓ | very high | \$ ~ \$\$\$ |

Tabella 3.1: Confronto tra sistemi di mocap full-body

4. Motion Capture Facciale

Anche per la registrazione dei movimenti facciali si può ricorrere a metodi con o senza markers. La cattura delle espressioni genera una libreria di espressioni che risulta essere molto comoda in fase di animazione, garantendo dei risultati più naturali rispetto a pose con keyframe: è importante che il modello abbia una densità sufficiente di vertici sul viso, a seconda del grado di realismo che si vuole raggiungere, la libreria di pose che si va a costruire rappresenta appunto le coordinate dei vertici che vengono deformati in fase di registrazione.

Nell'ambito delle tecnologie marker-based, il tracking avviene applicando sul viso dell'attore un numero variabile di marker soprattutto in quelle zone di maggiore deformazione dei muscoli facciali. La registrazione viene effettuata con camere ad alta risoluzione poste su degli appositi caschi modificati perchè la camera sia posta di fronte al viso dell'attore, e ne segua sempre i movimenti muovendosi sul suo stesso asse. Esagerando con il posizionamento dei marker si può incorrere in impedimenti per l'attore, derivando così una performance non ottimale o meccanica che si riflette in movimenti snaturati dell'avatar digitale al quale saranno applicati. In alternativa ai marker si possono valutare diversi approcci che seguono le regole della computer vision, algoritmi di ultima generazione che sono in grado di tracciare le pupille e le occlusioni dell'apparato dentale, ma la qualità di queste registrazioni dipende dalla risoluzione dell'attrezzatura a disposizione.

Nel film *Avatar* si è fatto ricorso a un sistema misto: i marker sul viso degli attori erano dei semplici riferimenti dipinti, registrati da camere ad alta risoluzione che utilizzarono algoritmi di elaborazione delle immagini per mostrare in real-time come output delle camere virtuali l'avatar digitale opportunamente animato.

Totalmente diverso fu l'approccio per '*Il Curioso Caso Di Benjamin Button*', dove la produzione decise di abbandonare qualsiasi tecnica di motion capture allora utilizzata nel cinema perché non ritenevano minimamente soddisfacente il risultato per il viso di Brad Pitt durante le fasi di vita del suo personaggio Benjamin. Prendendo come punto di partenza l'assunzione del F.A.C.S (Facial Action Coding System) introdotta nel 1978 da Ekman and Friesen secondo cui è possibile ricostruire ogni espressione facciale da un centinaio di unità singole definite, denominate *action unit* (AU), studiarono come estrapolare tutte le informazioni dei micro-movimenti dei muscoli facciali e applicarli ai vari modelli di Benjamin indipendentemente dall'età.



Figura 4.1: Tecnica del countouring come mostrata alla conferenza TED's 'How Benjamin Button got his face' (2009)

Grazie alla tecnica del *countour*, qui mostrata in Figura 4.1, ponendo l'attore di fronte a un sistema di camere (invece che applicare un sistema di marker sul viso) è stato possibile registrare con assoluta precisione tutte le espressioni del viso del soggetto registrato ad ogni singolo frame. Tramite l'utilizzo di questa tecnica è stata creata una libreria più che completa di espressioni di Brad Pitt da applicare al personaggio di Benjamin in qualsiasi stadio della sua vita.

Se si sposta lo sguardo al mondo dei videogiochi degli ultimi anni, il motion capture facciale sta raggiungendo dei risultati impressionanti in termini di realismo, e un esempio recente è il titolo 'Uncharted 4', pubblicato da Naughty Dog come esclusiva Sony Playstation 4 nel 2016. E' stato il primo capitolo della saga a introdurre delle head-cam per la registrazione delle espressioni facciali, prima di allora i volti erano animati a mano. Hanno investito in software proprietari per il rendering e il risultato è un videogioco completamente in cinematics che ha alzato definitivamente il livello di realismo che l'industria è in grado di raggiungere. Solo nel mese scorso hanno nuovamente superato il loro limite con la presentazione del trailer per 'The Last Of Us - Parte 2' confermandosi come la colonna portante dell'evoluzione della grafica nel videogioco, con particolare attenzione alla resa iperrealistica dei volti dei personaggi.



Figura 4.2: Motion Capture facciale per i personaggi di *Uncharted 4*, © *Naughty Dog*.
(cliccare sull'immagine per andare al video)

5. Pipeline per il Motion Capture

Indipendentemente dal sistema adottato, una tipica pipeline per il motion capture comprende diversi stadi, tra cui una fase di pianificazione delle riprese e degli assets da utilizzare che esula in parte dal settaggio del sistema di mocap, una fase di registrazione dei dati e successivamente la loro elaborazione e trasposizione sul modello target. Nella seconda parte, discutendo della pipeline del progetto *Reverie Dawnfall*, si esaminerà la pipeline adottata per la produzione del teaser trailer animato e come il motion capture si è inserito all'interno di essa.

Entrando invece nel dettaglio della produzione di motion capture vera e propria si può vedere come la rappresentazione della Figura 4.1 mostri alcuni dei momenti standard per la registrazione e post-produzione dei dati. Come è stato più volte ripetuto, l'output e l'utilizzo ottimale dei dati dipende dall'accuratezza del sistema, quindi alla base del processo vi è un momento di calibrazione, diverso per ogni sistema, ma immancabile per ogni tipologia di mocap.

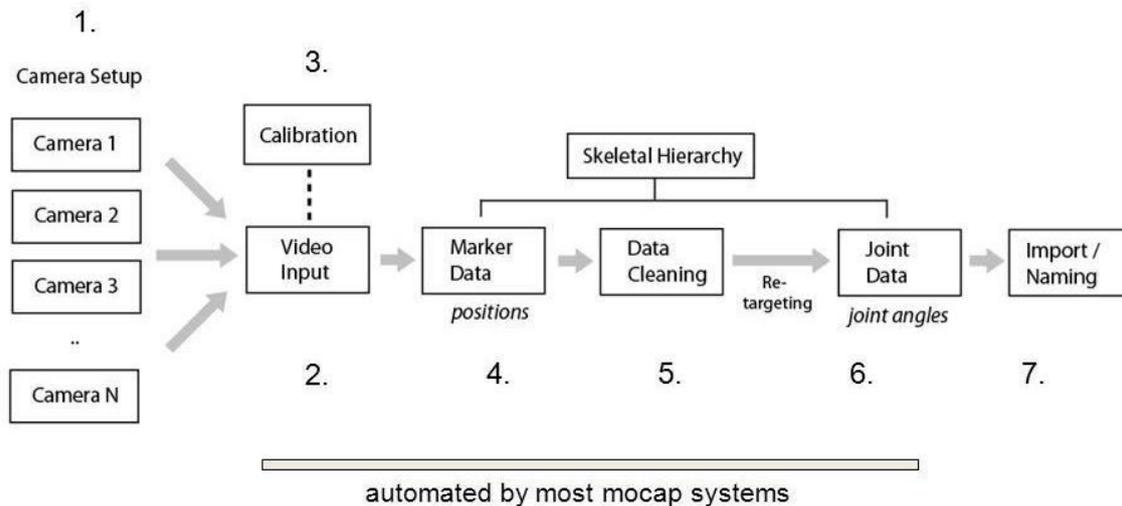


Figura 5.1: Esempio di pipeline per il motion capture

Prima di arrivare sul set e cominciare le riprese con gli attori, bisogna studiare il personaggio in CGI che sarà il target della registrazione. In particolare, nella costruzione del modello 3D è necessario prestare attenzione ad eventuali vincoli posti dal sistema di mocap che si implementeranno, impostare un corretto orientamento degli assi cartesiani durante il rigging in modo che i giunti siano orientati allo stesso modo e possibilmente adottare la stessa

convenzione del sistema mocap, e infine verificare che il numero di ossa sia corrispondente a quello generato dal sistema.

5.1 Calibrazione

La calibrazione comprende due tipi di analisi: in primo luogo si procede con la calibrazione del sistema di mocap, in secondo si passa alla calibrazione del soggetto. Come ci si può aspettare dai sistemi ottici, la calibrazione del sistema comprende tutti i controlli delle camere in uso nell'ambiente di cattura con l'ausilio di alcuni marker posti su corpi rigidi. Si definisce l'orientamento delle camere nello spazio relativamente alla loro posizione, e da questo momento è possibile triangolare la posizione dei marker che vengono visti da almeno due strumenti di ripresa.

Nei sistemi markerless questo procedimento viene meno a favore di altri momenti di calibrazione: per i sistemi magnetici si verificano i livelli di eventuali interferenze presenti nell'ambiente per ridurli al minimo possibile, mentre per i sistemi inerziali si verifica il corretto funzionamento del dispositivo wireless.

Calibrare il soggetto significa definire la sua posizione iniziale nello spazio e avviene ogni qualvolta si procede alla cattura dei movimenti. Si invita l'attore a stare in piedi nella consueta *T-pose* mentre il software di mocap calcola le relazioni tra i marker posti sull'attore e i riferimenti del sistema, in modo da trovare la migliore corrispondenza tra i marker e lo scheletro usato per visualizzare la simulazione sul compilatore. Il procedimento richiede pochi secondi ed è consigliato ripetere questo passo più volte durante le giornate di ripresa per assicurarsi che i sensori siano sempre allineati e funzionanti.

Per i sistemi inerziali la calibrazione del soggetto è un elemento fondamentale perché in questo momento si definisce la posizione relativa al punto 0 del terreno al quale il soggetto è poggiato: poiché non esiste un metodo automatico per valutare la posizione globale del campione, tutte le posizioni successive verranno elaborate dalla prima posizione registrata.

5.2 Performance capture

In questa fase si iniziano ad elaborare i dati veri e propri del motion capture e funziona esattamente come una consueta fase di ripresa. Nei sistemi non ottici si mostra il vero punto di forza di questa tecnologia, ovvero la possibilità di visualizzare in real-time il risultato della

performance capture. Ogni sistema che porta con sé un software dedicato permette di visualizzare un manichino standard al quale vengono incollati i movimenti generati dall'attore: se si vuole visualizzare direttamente i movimenti sul proprio personaggio in CGI occorre prima effettuare un processo di retargeting, cioè creare una corrispondenza tra lo scheletro del sistema mocap e quello proprio del personaggio. A questo proposito esistono software dedicati come Motion Builder di Autodesk, oppure plugin per i più moderni motori grafici come Unreal Engine e Unity.

Generalmente i dati prodotti dalla performance capture necessitano di opportuna elaborazione ma poter vedere in anteprima i movimenti sui propri personaggi è una risorsa importante per poter iniziare a valutare le modifiche opportune.

5.3 Elaborazione e pulizia dei dati

Una volta ottenuti i dati della performance capture inizia la fase di post-processing, che comprende l'analisi dei dati prodotti dai marker e la pulizia dei dati numerici. Per i sistemi ottici in questa fase si effettua il calcolo della componente rotazionale per ottenere dei dati completi e utilizzabili in fase di animazione. Per tutti i sistemi in esame, si svolgono le operazioni di valutazione qualitativa dei dati e recupero di eventuali gap dovuti ad errori di trasmissioni od occlusioni dei marker, e si procede con la pulizia delle curve numeriche prodotte lungo gli assi di rotazione e traslazione al fine di ottenere delle animazioni regolari e senza scarti visibili a occhio nudo.

Durante questo processo si può scegliere di lavorare sulla pulizia dei dati direttamente sul modello al quale verranno applicati tramite retargeting, oppure pulire precedentemente tramite la visualizzazione del solo scheletro mocap dipendentemente dal software di elaborazione che si intende utilizzare. Uno dei problemi comuni che si presenta in questo stadio e che è possibile risolvere con meccanismi semiautomatici di Motion Builder è lo *sliding* degli arti inferiori del soggetto sul terreno, un fenomeno che può avvenire durante lunghi momenti di cattura o dovuto alla scarsa accuratezza del sistema.

5.4 Applicazione del movimento

Se non si è fatto nel punto precedente, la fase di retargeting è il prossimo punto sulla lista. Con il retargeting è possibile assicurarsi che il modello in CGI di destinazione sia in grado di prendersi in carico di tutte le animazioni registrate con il sistema di motion capture. Una volta associato lo scheletro di mocap a quello del personaggio, si avrà piena libertà

sull'importazione di nuovi movimenti registrati anche in un secondo momento fintanto che non sarà modificata la gerarchia e la nomenclatura delle ossa dello scheletro sorgente, in questo modo per esempio in Maya sarà sufficiente fare un *drag & drop* del nuovo file di animazione nel progetto perché esso vada a sostituire lo scheletro mocap già presente e visualizzare il nuovo movimento. In una situazione in cui si lavora in real time invece, il retargeting è stato già effettuato alla base della pipeline quindi il personaggio vedrà già applicate le animazioni e approssimativamente l'unico problema da risolvere sarà quello descritto nella fase precedente ovvero la pulizia dei dati prodotti.

Ponendo la giusta attenzione dalla fase di preliminare fino a quella di post-processing è possibile restringere al minimo le correzioni intermedie su cui dover intervenire nelle fasi intermedie del progetto: una buona calibrazione iniziale dei sensori assicurandosi che siano simmetricamente allineati sul soggetto permetterà di ottenere una buona animazione globale del personaggio in CGI, senza dover ricorrere a riallineamenti delle ossa per far sembrare dritte delle spalle che in fase di registrazione erano visibilmente sfasate.

6. Case Study: Rokoko

Rokoko è un progetto nato nel 2012 nel contesto universitario della National Film School of Denmark, dall'idea di Jacob Balslev di voler creare un teatro d'animazione dal vivo. Per poter realizzare il progetto volevano avvalersi della motion capture, totalmente inaccessibile per degli studenti, e nei primi due anni si occuparono di ricercare e sviluppare una tecnologia che potesse essere accessibile *democraticamente*. Nel 2015 hanno iniziato a testare il primo prototipo di Salto, oggi Smartuit Pro, durante eventi e manifestazioni di tecnologia e animazione riscuotendo un notevole successo, che li ha portati ad aprire una campagna Kickstarter nell'Ottobre dello stesso anno, con l'obiettivo di consegnare ai 'backers' (coloro che investono nelle campagne di crowdfunding in cambio di premi che corrispondono molto spesso al prodotto che stanno supportando) la tuta completa e funzionante entro il primo trimestre del 2016. Dalla timeline di produzione reperibile sulla pagina Kickstarter di Salto, si può notare come abbiano iniziato a prototipare un modello di guanti già dai primi stadi di sviluppo, pezzo attualmente mancante della tuta e attualmente in programma per il primo trimestre del 2018. La campagna di crowdfunding viene lanciata ufficialmente il 22 Settembre 2015, annunciata come la rivoluzione della motion capture, e in un solo mese riesce a superare la soglia di 100,000\$, requisito della compagnia per iniziare una produzione, raggiungendo di fatto con 223 backers un totale di 118,534\$.

Il 18 Novembre 2016 viene ufficialmente presentata la Smartsuit Pro, e le spedizioni delle prime tute sono iniziate a fine primavera 2017, più o meno lo stesso periodo in cui è iniziata questa ricerca. La scelta di questa tuta è stata una sfida ad occhi chiusi, in quanto si trattava di un prodotto nuovo che doveva ancora fare il suo ingresso nel mercato, a differenza dell'esoscheletro Neuron della cinese Noitom, e quindi non sono ancora presenti dei feedback di altri utenti. D'altro canto si è rivelata essere un'occasione unica, in quanto ci ha permesso di entrare in contatto con la realtà danese e stabilire un rapporto di collaborazione con il Politecnico di Torino e con lo studio Robin, dove è in produzione il progetto Reverie Dawnfall.

Alla fine di Luglio abbiamo contattato Matias Søndergaard e Mikkel Lucas Overby, rispettivamente co-founder di Rokoko e COO, e abbiamo aperto un dialogo sulle possibilità di collaborazione all'interno dei progetti universitari e dei loro piani di sviluppo: le proposte più interessanti sono state sicuramente il beta-testing dei guanti per il mocap delle mani, e lo sviluppo di una libreria di motion capture simile a quella di Mixamo ma dedicata ai mocap

data creati con la Smartsuit Pro. Da una interessante discussione e scambio di opinioni è scaturito un viaggio che mi ha permesso di visitare la sede centrale di Copenaghen (ne hanno aperta da poco un'altra a San Francisco) e di fare del training direttamente da chi la tuta l'ha ideata e creata. Lo studio è di per sé un piccolo laboratorio, i sensori vengono assemblati e testati mentre gli sviluppatori più avvezzi all'informatica svolgono le operazioni di calibrazione e problem solving. Attualmente hanno all'attivo una produzione di più di 1000 pezzi pronti per la spedizione, e non è strano trovare tra le scrivanie delle schede madre o degli hub di connessione. Stanno continuando a sviluppare e a migliorare la tecnologia, per poter trovare delle soluzioni che permettendo di abbassare ulteriormente il prezzo del prodotto finale, attualmente di 2495\$, circa 2100€.

6.1 Smartsuit Pro e Smartuit Pro Studio

La Smartsuit Pro viene descritta come un intero studio di motion capture markerless, è composta da 19 sensori IMU (Inertial Measurement Unit) a 9 gradi di libertà connessi a un hub centrale che contiene la scheda di memoria e permette anche una connessione WiFi a 2,4 o 5 GHz di banda. L'unico vincolo fisico è rappresentato dalla taglia del tessuto, quindi la taglia acquistata si adatta a un certo range di attori in altezza e proporzioni, ma è data la possibilità di acquistare un secondo tessuto di taglia diversa per poter avere un ricambio oppure più libertà di utilizzo fra persone.

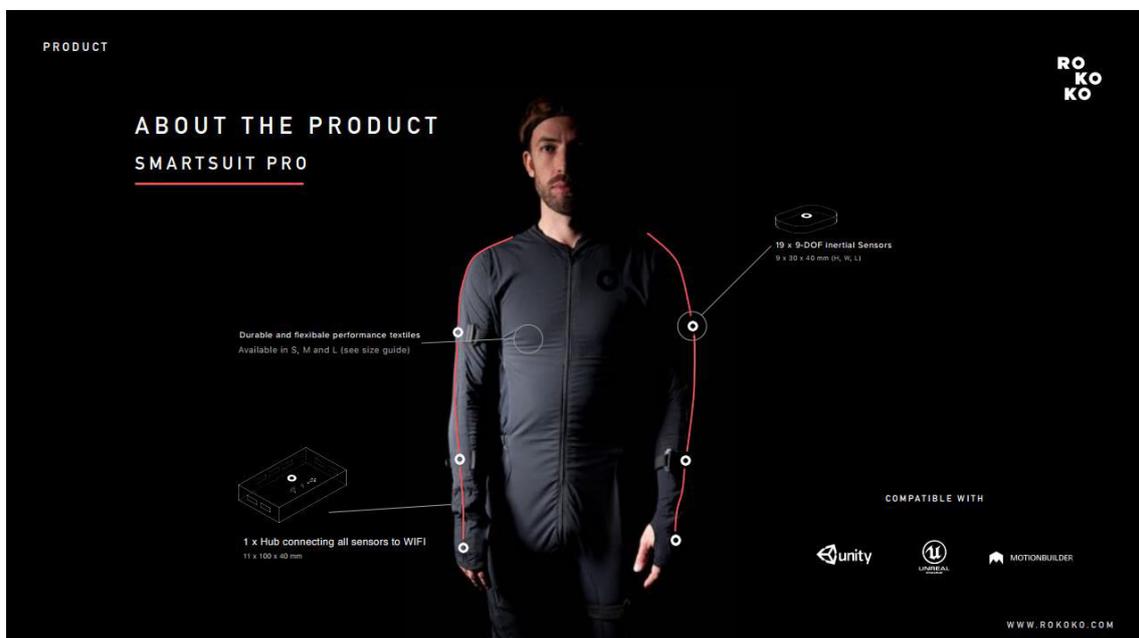


Figura 6.1: Scheda di presentazione della tuta Smartsuit Pro

Insieme alla tuta è stato sviluppato un software per poter visualizzare e registrare dal vivo la performance dell'attore. All'interno del software è possibile creare dei profili specifici per ogni persona, impostando le diverse informazioni quali l'altezza, l'ampiezza delle braccia e delle spalle, la lunghezza dei piedi e lo spazio tra le due gambe: in questa maniera ogni volta che la tuta viene indossata, si può avere una perfetta corrispondenza tra il modello visualizzato sullo schermo e chi sta indossando la tuta. L'interfaccia è intuitiva e si può scegliere se registrare la performance in unicast o in broadcast, questo significa che uno o più computer possono visualizzare contemporaneamente la tuta in attività: si possono connettere al massimo cinque tute in contemporanea e ognuna può essere gestita da un profilo dedicato, ad esempio ogni computer può ricevere in broadcast una sola tuta in modo che più persone possano lavorare insieme su un soggetto in contemporanea, mentre tutti gli attori vengono registrati. Una piccola schermata monitora in tempo reale che tutti i sensori siano funzionanti e non ci siano interferenze di tipo magnetico: quando un sensore rischia di subire interferenze il marker si illumina di giallo o di rosso a seconda della gravità dell'interferenza, quando tutti i marker sono verdi ci si trova nella condizione ottimale di lavoro. La tuta viene consegnata già completamente assemblata e funzionale, l'unica accortezza prima di registrare una performance è assicurarsi che i marker delle spalle, dei fianchi e delle altre zone simmetriche legate alla spina dorsale allineati e posizionati allo stesso modo.

Si può suddividere la scena in takes e organizzare intere giornate di lavoro in cartelle direttamente da programma, e i file si possono esportare sia in .BVH che in .FBX, formati universali che vengono letti da tutti i principali programmi che si occupano del 3D. Rokoko ha inoltre sviluppato dei plugin per Unity e Unreal che supportano la visualizzazione in realtime per poter connettere direttamente l'attore ad un alias digitale e poter quindi evitare l'utilizzo del loro software proprietario.

Questo tipo di tuta con sensori inerziali presenta attualmente un problema sostanziale, ovvero che non essendoci delle camere che calcolano l'esatta posizione della persona nello spazio, bisogna sempre definire manualmente tale impostazione: il software in dotazione ha un tasto per inizializzare la tuta e la posizione all'interno dello spazio. Un altro aspetto da tenere a mente è che allo stato attuale, i sensori non presentano un metodo di calibrazione globale, deve esserci sempre un contatto tra l'attore e lo spazio circostante, nel caso in esame la posizione relativa viene calcolata dai sensori degli arti inferiori: al momento quindi non si possono registrare azioni come un salto o altre situazioni che portano la tuta a non avere contatti con qualcosa di fisico dell'ambiente circostante senza riscontrare errori nella performance. A parte questo promemoria, la libertà d'azione che viene offerta da questa

tecnologia è enorme rispetto ai sistemi di motion capture ottici poiché non c'è un vincolo con l'ambiente di registrazione e si può letteralmente *fare mocap ovunque*. Il Wi-Fi della Smartsuit Pro supporta una distanza fino a 100m, rendendo possibile una registrazione accurata dei movimenti sia in ambienti chiusi che all'aperto. È fondamentale tuttavia che l'hotspot sia libero da altri device connessi in modo da garantire una performance di ripresa stabile, che si assesta generalmente sui 110fps con uno scarto massimo di 10fps su entrambi i fronti.

6.2 Performance test

Durante la permanenza a Copenaghen, ospite dello studio Rokoko, ho avuto modo di effettuare un primo test e di confrontare i risultati con il team di sviluppatori della tuta. La registrazione consisteva in un paio di minuti simulando dei semplici movimenti come la camminata, rotazione delle braccia e del collo e qualche piegamento sulle ginocchia. Per l'esportazione del file ho scelto il formato .FBX perché essendo già convertito è più facile da interpretare all'interno di Maya, in quanto il formato .BVH è un formato ormai obsoleto e molto spesso richiede un passaggio intermedio per essere leggibile dai programmi di elaborazione 3D. Importato il file in Maya viene visualizzato uno scheletro contenente tutte le informazioni di movimento che sono state registrate: lo scheletro è già suddiviso e organizzato in maniera gerarchica secondo lo standard dello *skeletal data*. Avere un nodo al di sopra del giunto dei fianchi agente come reference per tutto il sistema permette di effettuare trasformazioni al modello senza intaccare i dati del mocap registrato.

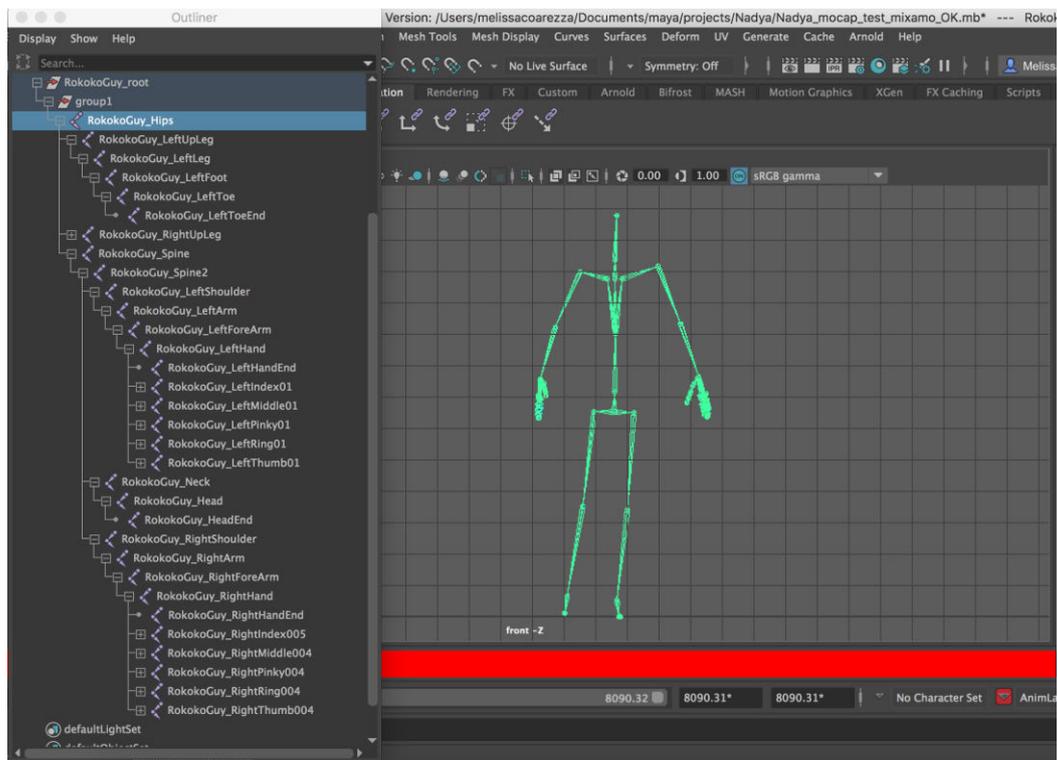


Figura 6.2: Gerarchia dei giunti estratti dallo *skeletal data* della tuta Rokoko: *RokokoGuy_Hips* è la radice effettiva dello scheletro, da cui si snodano come *children* le gambe e la spina dorsale.

Il rig generato dalla tuta viene associato a quello del personaggio tramite un processo di retargeting, in linea teorica dopo aver selezionato le tracce di interesse e dopo aver pulito i dati del mocap. La curiosità ha vinto sulla regola e come primo test ho deliberatamente invertito i passaggi per avere un feedback su come avrebbe reagito il sistema sul personaggio che era stato precedentemente creato in Maya e riggato in due versioni: una con l'ausilio di Mixamo (un software online acquisito da Adobe e recentemente in via di smantellamento), e un'altra con l'utilizzo di uno script denominato *'Advanced Skeleton'* che permette di creare con diverse alternative dei rig complessi comprensivi anche degli arti delle mani e la possibilità di creare anche un rig per il viso. In entrambi i casi, il retargeting si è rivelato efficace nonostante la creazione manuale dei constraints tra i due scheletri che ha causato qualche discrepanza sull'asse y del sistema del personaggio in esame. In realtà, tramite Motion Builder o alcuni strumenti di Maya dedicati al rigging, è possibile evitare qualsiasi tipo di problema legati alle differenze di scala tra i due scheletri, ma come primo esperimento era più che sufficiente avere una corrispondenza esatta almeno nel movimento delle ossa. Riassumendo, i passi nello stadio del retargeting manuale sono stati i seguenti:

- 1) *Orient Constraint* tra i giunti dello scheletro mocap e il rig del personaggio, iterando il processo per ogni coppia di giunti, in modo da orientare gli assi di rotazione locale dei giunti identicamente per entrambi gli scheletri.
- 2) *Point Constraint* tra i due rig in modo che il target seguisse la posizione del rig sorgente.

Con questi semplici passaggi, il personaggio di Nadya è stato animato dal rig mocap e il risultato è stato discretamente positivo. Il compito successivo ha previsto l'analisi più approfondita dei dati prodotti dalla performance capture: in Figura 6.2 viene riportato il grafico dei dati grezzi ottenuti dalla registrazione del movimento per il giunto del braccio sinistro. I dati di tipo *skeletal* contengono informazioni di rotazione e traslazione, a differenza dei *markers data* i quali contengono solamente la componente traslazionale, ma la componente di traslazione è presente solo nel giunto *root*, nel nostro caso localizzato in *RokokoGuy_Hips*. Il grafico di questi dati mostra come le componenti di rotazione sui 3 assi richiedano un intervento massiccio perché, oltre ad essere molto densi come dati, sono presenti molti salti nelle funzioni che richiedono di essere interpretati e interpolati correttamente.

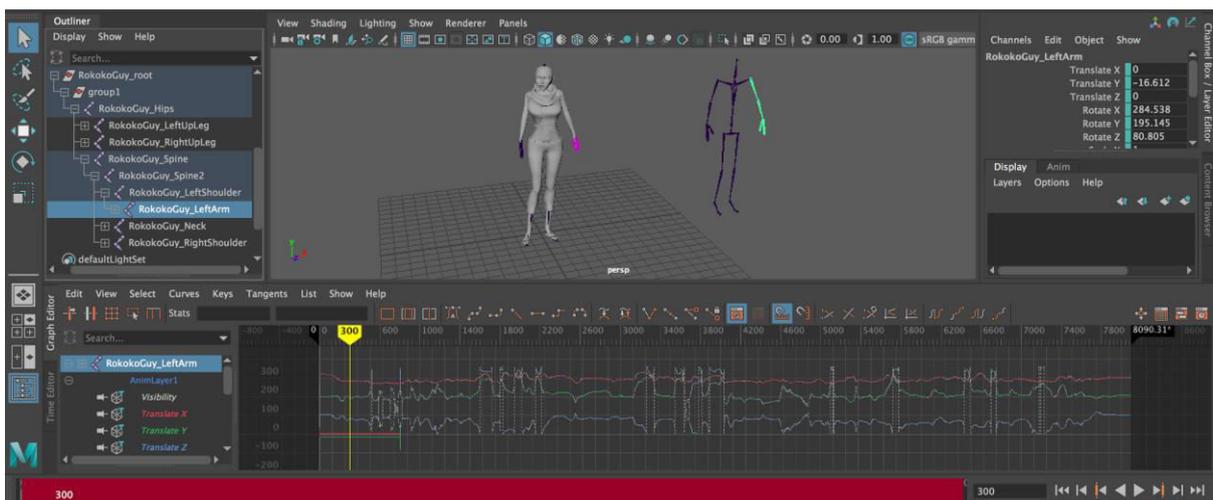
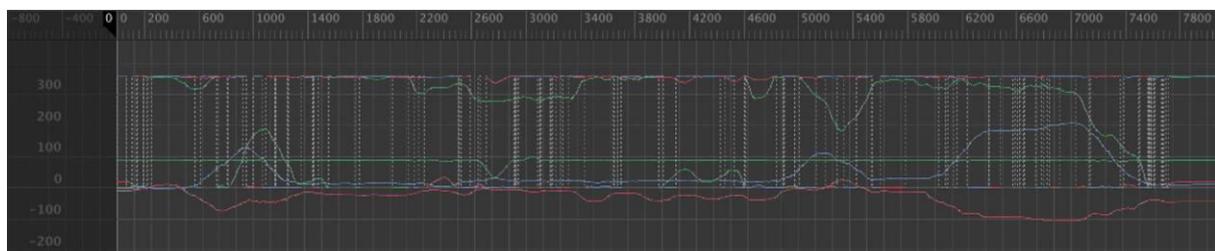


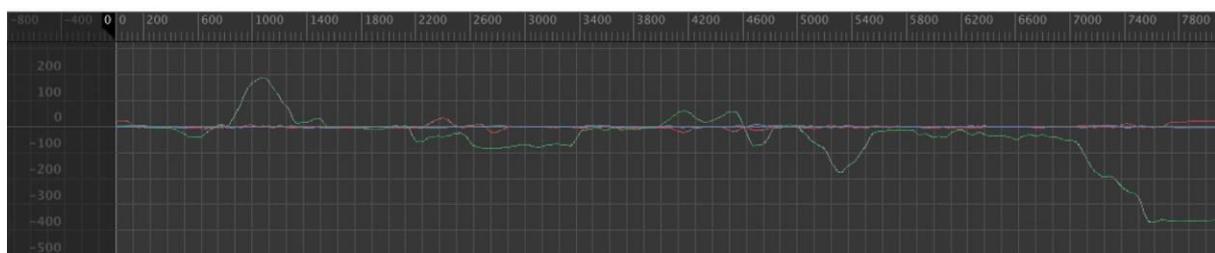
Figura 6.3: Graph Editor relativo al giunto *RokokoGuy_LeftArm* prima del cleaning dei dati.

E' buona norma applicare inizialmente ai dati un *filtro di Eulero*, specie nei casi in cui ci si trova davanti a un grafico come in Figura 6.3: questo filtro serve a prevenire la rotazione oltre i 360°, e la sua sola applicazione porta a un miglioramento sostanziale del grafico globale. Dopodiché si inizia a valutare il grafico su ogni asse preso singolarmente alla ricerca di

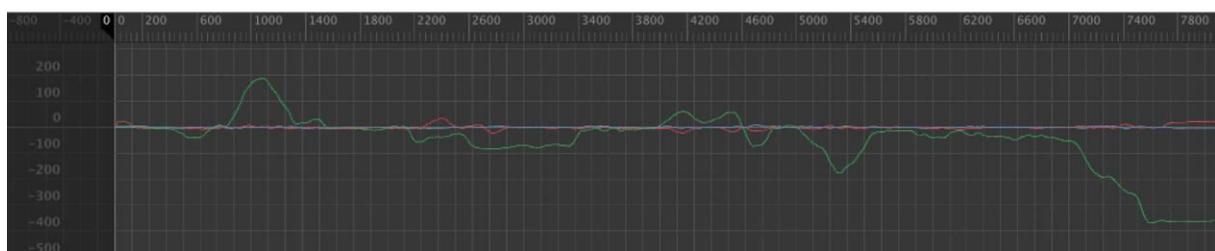
spikes, singoli keyframe scappati all'andamento del grafico generale che possono generare rumore nell'animazione se non trattati correttamente. Campionare una curva che non è stata prima verificata può produrre delle funzioni diverse da quelle della fonte originale perchè gli *spikes* vengono interpolati con delle spline quindi dopo il ricampionamento una curva apparentemente corretta renderà gli errori più difficili da individuare.



(a) sequenza originale



(b) Filtro di Eulero applicato



(c) Ricampionamento della curva

Figura 6.4: Step per la pulizia dei dati mocap: (a) sequenza originale, (b) applicazione del filtro di Eulero e pulizia degli *spikes*, (c) ricampionamento della curva

Nel caso particolare in analisi, al frame 804 si notava un flickering ad entrambe le braccia, un movimento non registrato con la tuta e quindi frutto di un errore nei dati elaborati in output. Valutando il grafico per i giunti dalla spalla in giù intorno a quei frame dove è stato riscontrato l'errore è stato possibile risalire all'origine del dato corrotto. Le Figure 3.4 e 3.5 mostrano la situazione problematica, che coinvolge due assi di rotazione, -z e -y. Lavorando su un solo asse alla volta il problema sembrava peggiorare, mostrando una rotazione impazzita della durata di un frame; applicando il filtro di Eulero e ricampionando la curva portava allo stesso tipo di peggioramento. È stato necessario intervenire manualmente,

scalando il valore del frame per entrambi gli assi invece è stato possibile risolvere quel piccolo errore visibile a colpo d'occhio.

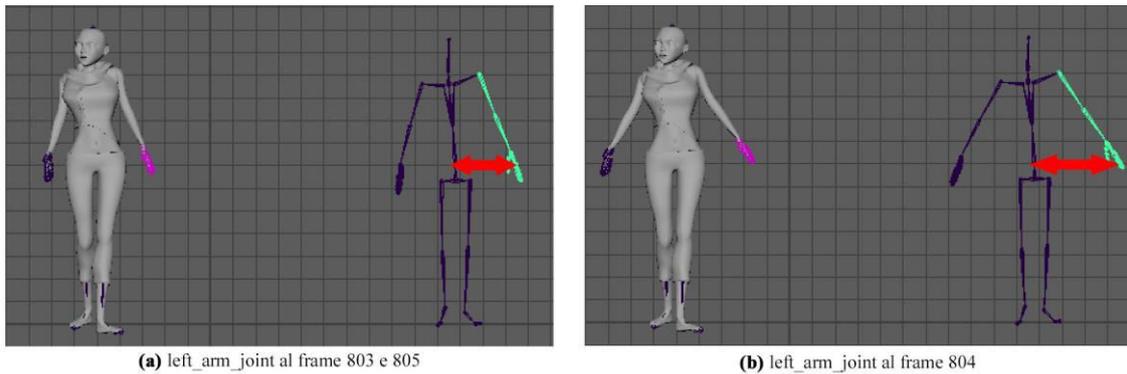
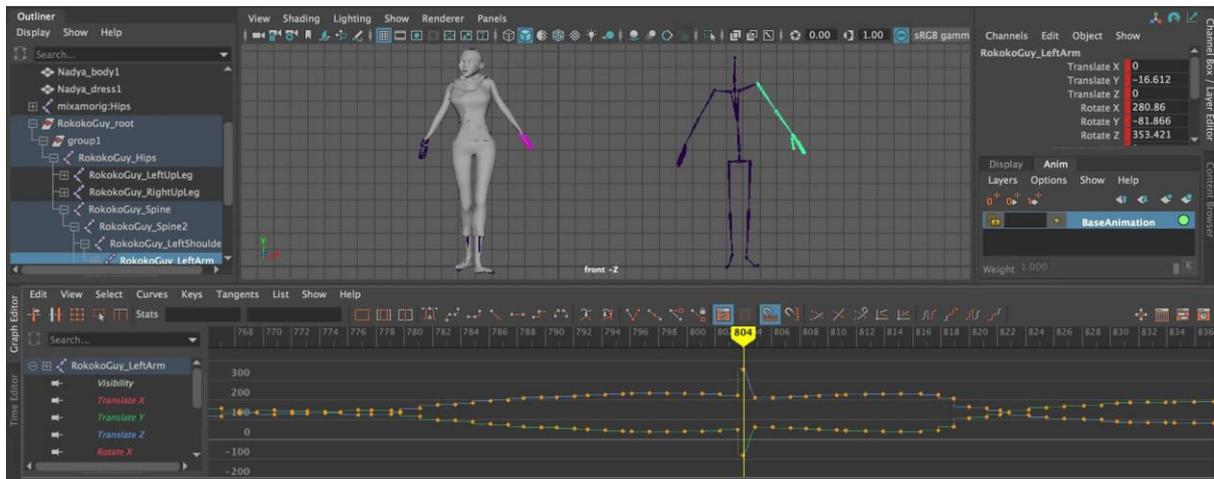
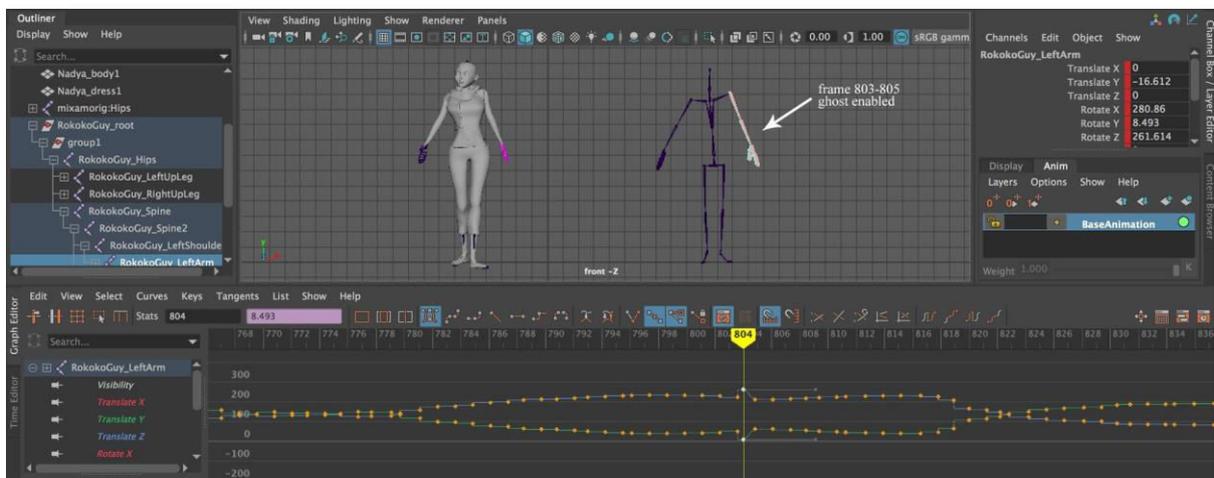


Figura 6.5: Rotazione delle braccia nei 3 frame esaminati dove si è riscontrato il flickering a occhio nudo



(a)



(b)

Figura 6.6: Analisi del grafico (a) e correzione dei keyframe errati (b). Dal graph editor è stato possibile individuare l'errore dei dati e attivando la visione *ghost* si è potuto correggere la rotazione dell'osso fino alla posizione corretta.

La conclusione di questo primo esperimento, e primo approccio verso la tecnologia di motion capture, ha dimostrato come non esista un'unica via da seguire durante l'elaborazione dei dati prodotti dalla tuta inerziale ma è possibile, e consigliabile, studiare il problema che si ha davanti per poter ragionare in maniera mirata sulla migliore soluzione al problema. Sicuramente accumulando esperienza nell'utilizzo della tecnologia la velocità di ragionamento e di risoluzione dei dati rumorosi diventerà un processo sempre più veloce ma è stato possibile ipotizzare alcuni punti in comune a tutte le strategie adottate studiando il modello che si potrebbero tradurre nella creazione di piccoli script per automatizzare alcuni passi della pulizia dei keyframe. Durante la permanenza negli studi di Copenaghen è stato possibile confrontarsi con gli sviluppatori sul lavoro operativo effettuato sui dati, per migliorare le prestazioni del loro prodotto, in particolare è stata ritenuta interessante l'idea di applicare ai dati prodotti dalle registrazioni un filtro di Eulero preventivo, per poter offrire agli animatori dei dati in output più corretti. Alla fine di questa ricerca è stata pubblicata una nuova versione del software, di cui si spiegheranno le caratteristiche a breve, ma si può notare come il dialogo con gli sviluppatori abbia contribuito al miglioramento delle prestazioni del programma, in quanto è stata inserita una funzione che permette di effettuare una levigatura dei dati in maniera semplice, prima di esportare il file della ripresa.

6.3 Formulazione del problema

Indagando ulteriormente i limiti di questa tecnologia sono stati svolti presso Robin Studio ulteriori test di performance analysis, ed è stata studiata più nel dettaglio la problematica del salto e di altre azioni che comportassero l'assenza di un contatto tra il punto 0 di calibrazione e i sensori dei giunti *foot*. Come accennato in precedenza, durante la fase di calibrazione viene chiesto al soggetto di restare fermo per qualche secondo mentre il software calcola la posizione 0 dalla quale verranno interpolati i successivi dati di traslazione e rotazione: che sia il pavimento oppure che sia il gradino sul quale il soggetto è salito, il programma di elaborazione riporterà sempre le coordinate [0,0,0] come punto di partenza nello spazio virtuale. Ciò detto, il soggetto sullo scalino sarà registrato dal programma come il soggetto sul pavimento in piano, non c'è una differenza sull'asse dell'altezza ed entrambi sembreranno sullo stesso piano.

Per la produzione del teaser trailer oggetto di questa tesi non è prevista alcuna delle azioni non permesse dal sistema, tuttavia è una condizione indispensabile poter lavorare con

piena libertà di movimento durante la produzione serializzata vera e propria, ed è necessario quindi investire del tempo per trovare una soluzione a un problema che inizialmente non era stato contemplato.

Il problema consiste nell'impossibilità di calcolare con precisione le coordinate globali dei sensori inerziali con gli strumenti messi attualmente a disposizione dal costruttore. In letteratura si scopre come esistano metodi matematici per bypassare il limite del sistema IMU, soprattutto nell'ambito della ricerca sportiva e clinica. Diversi studi, come [5, 6, 8] partono dall'analisi empirica del problema e da un'analisi del sistema IMU, come descritto sinteticamente nella sezione precedente di questa ricerca. Molto spesso [6] si ritrovano implementazioni dell'algoritmo di Kalman, un filtro stimatore ricorsivo ottimo, per poter correggere l'errore nella stima dell'orientamento confrontando i valori ottenuti dal sistema inerziale in combinazione con un sistema ottico (tipicamente Vicon).

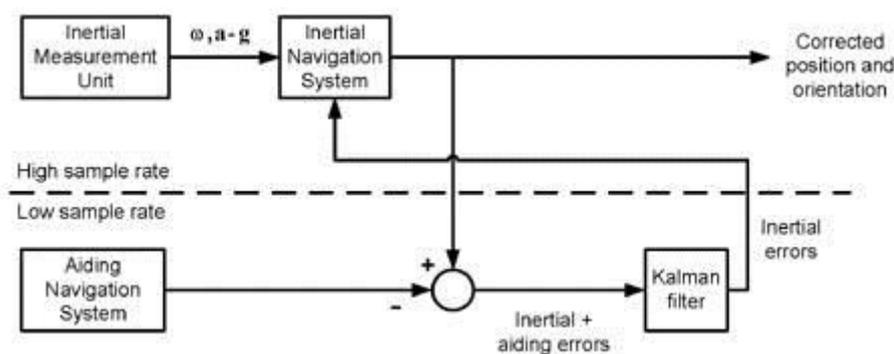


Figura 6.7: Sensor fusion con l'applicazione del filtro di Kalman per il calcolo della posizione globale

La struttura mostrata qui sopra rappresenta la struttura del filtro Kalman complementare per stima di posizione e orientamento che combinano misure inerziali e altre di supporto. I segnali ottenuti dall'IMU ($a-g$ e w) forniscono l'input per l'INS (Sistema di Navigazione Inerziale). Con la doppia integrazione delle accelerazioni, la posizione è stimata ad alta frequenza. A una frequenza che è la più bassa possibile, il sistema di supporto fornisce stime di posizione. La differenza tra le stime inerziali e di supporto viene fornita al filtro di Kalman. Sulla base del modello di sistema, che deve essere formato da funzioni lineare dello stato corrente, i filtri di Kalman stimano la propagazione degli errori. Le uscite del filtro vengono inviate indietro per correggere le stime di posizione, velocità, accelerazione e orientamento. Matematicamente questo modello è comparso più volte nella letteratura consultata, ed è stato quindi assunto come teoria più valida alla base dello studio del nostro problema.

Il filtro di Kalman è utile per combinare i dati di diverse misure indirette e rumorose. Non esiste un sensore perfetto, ogni tipo ha i suoi punti forti e deboli, e questo filtro è in grado di pesare correttamente le fonti di informazione, con la conoscenza delle caratteristiche del segnale in base ai propri modelli per sfruttare al meglio tutti i dati di ciascuno dei sensori. L'idea alla base del sensor fusion è che le caratteristiche di un tipo di sensore vengono utilizzate per superare i limiti di un altro sensore. Ad esempio, i sensori magnetici sono usati come riferimento per evitare che l'integrazione del giroscopio accumuli un errore di stima dell'orientamento intorno all'asse verticale. Tuttavia, il ferro e altri materiali magnetici possono disturbare il campo magnetico locale e, di conseguenza si ripropone lo stesso problema. Le caratteristiche spaziali e temporali dei disturbi magnetici saranno diverse da quelle relative agli errori di drift (una sistematica variazione di offset nel tempo) del giroscopio: è proprio usando questa conoscenza a priori che gli effetti del drift e dei disturbi possono essere minimizzati. I sensori inerziali del sistema di navigazione inerziale (INS) possono essere montati sui veicoli in modo tale da rimanere livellati e puntati in una direzione fissa. Questo sistema si basa su una serie di sospensioni cardaniche e sensori collegati su tre assi per monitorare gli angoli in ogni momento. Un altro tipo di INS è il sistema di strapdown che elimina l'uso di sospensioni cardaniche che è adatto per l'analisi del movimento umano: in questo caso, i giroscopi e gli accelerometri sono montati direttamente sulla struttura del veicolo o fissati sul segmento del corpo. Le misure sono fatte in riferimento agli assi locali di *roll*, *pitch* e *yaw*.

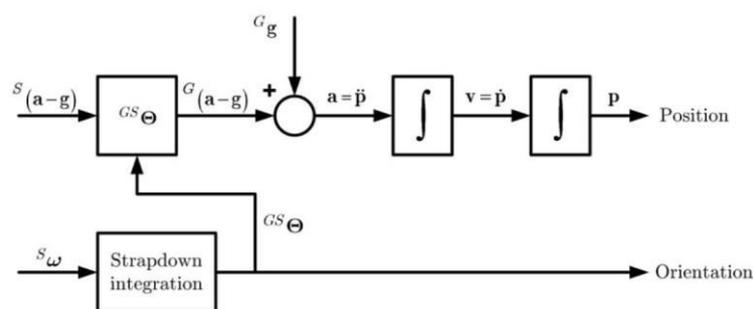


Figura 6.8: Sistema di navigazione inerziale (INU). La posizione globale viene ricavata integrando due volte il valore dell'accelerazione ricavata dall'IMU: s_ω è la velocità angolare del giroscopio mentre $a-g$ è il segnale dell'accelerometro.

Esistono quindi diverse ipotesi studiate nell'ambito della ricerca universitaria, ma è difficile reperire dei prodotti commerciali in grado di offrire una soluzione a basso costo, basati sui

risultati di queste ricerche. Si discutono quindi alcuni approcci per studiare delle soluzioni pratiche e convenienti al problema preso in esame, partendo sempre dall'assunto che si sta lavorando in un ambiente indipendente che molto spesso è sinonimo de '*l'arte del sapersi arrangiare*':

- 1) Seguire la via della ricerca e studiare a fondo il modello matematico reperibile in diverse versioni in letteratura, e ricavare manualmente dal sistema a disposizione i dati necessari per correggere la valutazione del dato globale;
- 2) Assumere come valide le ipotesi studiate dai colleghi ricercatori nell'ambito dell'elettronica e dell'informatica e costruire un modello semplificato che sia in grado di riprodurre come fine lo stesso risultato a fronte di un margine di errore conosciuto e accettabile.

La prima ipotesi è stata scartata a causa del processo altamente sperimentale e dispersivo in termini di tempo poiché richiederebbe mesi di ricerca e sviluppo sulla base di un processo *trial & error*, senza contare che bisognerebbe riuscire a reperire i dati prodotti dai sensori della tuta Rokoko e questo significa decodificare le librerie prodotte dall'azienda che non sono open-source. Contattando l'azienda inoltre, ci è stato assicurato che questo tipo di funzionalità sarà implementata, come attestano anche diversi video di presentazione del prodotto dove tutte queste azioni si rivelano possibili. La seconda ipotesi è sembrata più adatta al nostro ambiente e più sostenibile in un lasso breve di tempo, quindi le attenzioni si sono concentrate verso la ricerca degli strumenti che sarebbero serviti al raggiungimento del risultato.

6.4 Soluzione proposta e aggiornamento Smartsuit Studio

In attesa dell'aggiornamento della tuta si è voluta trovare una soluzione provvisoria che rispondesse alle esigenze di produzione. Il risultato delle ricerche ha dato vita alla creazione di un sistema ibrido a basso costo, che ha accostato alla registrazione della tuta inerziale un sistema ottico con l'utilizzo di quattro camere Playstation Eye: tre sarebbero sufficienti ma per avere una maggiore profondità di campo e quindi dei dati più sensibili se ne è aggiunta una. Lo stesso tipo di settaggio si può ottenere con l'utilizzo di almeno due Kinect. Il software di cattura del motion capture con questo tipo di sistema è iPi Soft, che offre un'ampia documentazione su quali siano le possibili configurazioni del sistema per poter registrare del motion capture completamente markerless.

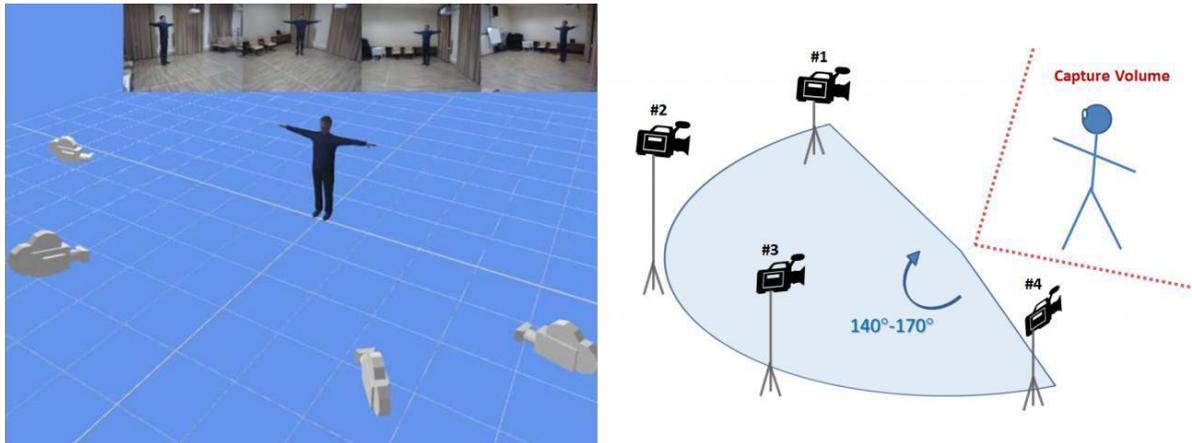


Figura 6.9: Configurazione del volume di cattura ideale per l'utilizzo di un sistema ottico di motion capture

L'idea alla base di questa soluzione consiste nella registrazione dell'attore con il sistema di telecamera disposte come in Figura 6.8, e contemporaneamente la cattura dei movimenti con la tuta inerziale. L'output di entrambi i dati verrà conseguentemente trasferito in Maya dove verranno unite le due animazioni registrate per ottenere il risultato finale fedele alla performance attoriale.

Prima ancora di riuscire a testare il sistema ibrido nella produzione vera e propria del prodotto animato, è stato annunciato un aggiornamento corposo che ha apportato un miglioramento fondamentale per la tuta Smartsuit e il suo programma di accompagnamento Smartsuit Studio, cioè l'introduzione dei filtri seguenti:

- *Locomotion* - questo filtro ricalcola il movimento guidando la traslazione e la velocità del corpo in base all'orientamento delle articolazioni e stima un nuovo comportamento di bloccaggio della mano e del piede. Consente inoltre il feedback dell'utente per regolare il blocco dei piedi e delle mani. Questo filtro migliora la locomozione e introduce il **salto**, la corsa e la possibilità di stare sulle mani (es. compiere una verticale) sia durante lo streaming dal vivo che durante la riproduzione.
- *Drift fix* - questo filtro ricalcola il movimento del corpo che l'utente fornisce tramite una posizione iniziale e finale per il movimento. Questo filtro può correggere errori di drift nel movimento di Smartsuit assicurandosi che la tuta inizi dal punto di partenza corretto e finisca sempre nel punto finale desiderato. Questo filtro funziona solo su registrazioni e non su live streaming.
- *Toe Bending* - questo filtro modifica le dita dei piedi in modo che non penetrino nel terreno. Funziona sia su registrazioni che live streaming.

- *Tapis roulant* - questo filtro blocca la posizione di Smartsuit su un piano o su un punto, a seconda delle impostazioni specificate dall'utente nel filtro. Funziona sia su registrazioni che live streaming.

Il filtro più importante tra quelli elencati è sicuramente quello relativo alla locomozione, perché introduce finalmente la risoluzione alla caratteristica problematica che si è studiata analiticamente, la possibilità di saltare. Grazie a questo aggiornamento, che seppur ancora in fase sperimentale si dimostra essere davvero promettente, il motion capture inerziale può finalmente sfruttare appieno le sue potenzialità e la più totale libertà di movimento, una libertà che si sarebbe dovuta limitare con la costruzione di un sistema ibrido.

7. Case Study: Faceware Technologies

Faceware Technologies è nata nel 2012 da Image Metrics, azienda fornitrice di tecnologie per l'animazione facciale e motion capture nel settore dell'intrattenimento (fondata nel 2000). Nel 2008, Image Metrics presentò una versione beta della sua tecnologia di animazione facciale agli studi cinematografici e di effetti speciali, mostrando la performance di un attore catturato in video e rimappato sul modello in CGI. La demo suscitò immediatamente l'interesse delle aziende poiché offriva l'occasione di incorporare all'interno della pipeline interna l'animazione facciale con motion capture, invece che rivolgersi alla ditta esterna. Nel 2010, Image Metrics ha lanciato la piattaforma tecnologica per l'animazione facciale Faceware focalizzata sull'aumentare il controllo creativo, l'efficienza e la velocità di produzione per gli animatori. Il software può essere integrato in qualsiasi pipeline o utilizzato con qualsiasi motore di gioco. Il primo studio a firmare come cliente di Faceware è stato Bungie, studio creatore del successo videoludico Destiny, che ha incorporato il software nella sua produzione interna. Negli anni successivi, la Image Metrics raccolse 8 milioni di dollari di finanziamenti e dichiarò il software Faceware come unica linea di business, presentando il lancio di Faceware 3.0 tramite un workshop e una presentazione al CTN Animation Expo 2011 intitolato "*Faceware: Creating an Immersive Experience through Facial Animation*". L'aggiornamento includeva l'auto-posa, un database di posa condiviso e il perfezionamento della curva, funzionalità che sono oggi alla base delle potenzialità del programma.

Image Metrics ha venduto ufficialmente il software Faceware alla sua società consociata, Faceware Technologies, nel gennaio 2012, e da quel momento l'azienda si è concentrata sulla produzione e distribuzione della sua tecnologia agli animatori professionisti. Attualmente i software proprietari di Faceware Technologies in commercio sono Faceware Analyzer, Faceware Retargeter, e la versione Live che rappresenta la fusione dei due programmi da utilizzare in real-time. Durante lo sviluppo di questa ricerca sono riuscite ad ottenere una licenza di *personal learning*, integrando ufficialmente nella pipeline di progetto anche il motion capture facciale con il software commerciale più in uso nell'industria. Per evitare ridondanze nella stesura di questo testo, l'utilizzo dei programmi verrà ampiamente trattato nella seconda parte, durante l'animazione facciale e il retargeting della performance capture della nostra attrice. Le sezioni successive saranno dedicate a una descrizione delle funzionalità e delle soluzioni offerte dai programmi in uso.

Navigando sul loro sito web si può vedere come vengano offerti prodotti sia dal lato software che da quello hardware, in particolare Faceware fornisce dei sistemi composti da elmetti specifici per il motion capture e camere GoPro. Questo input ha avviato una ricerca interna per poter costruire lo stesso tipo di sistema ma a basso costo, in quanto già in possesso della camera suggerita dal sito. Con gli accessori della GoPro e un comune casco debitamente imbottito, siamo riusciti a creare ad un costo irrisorio un sistema di cattura per la performance facciale.

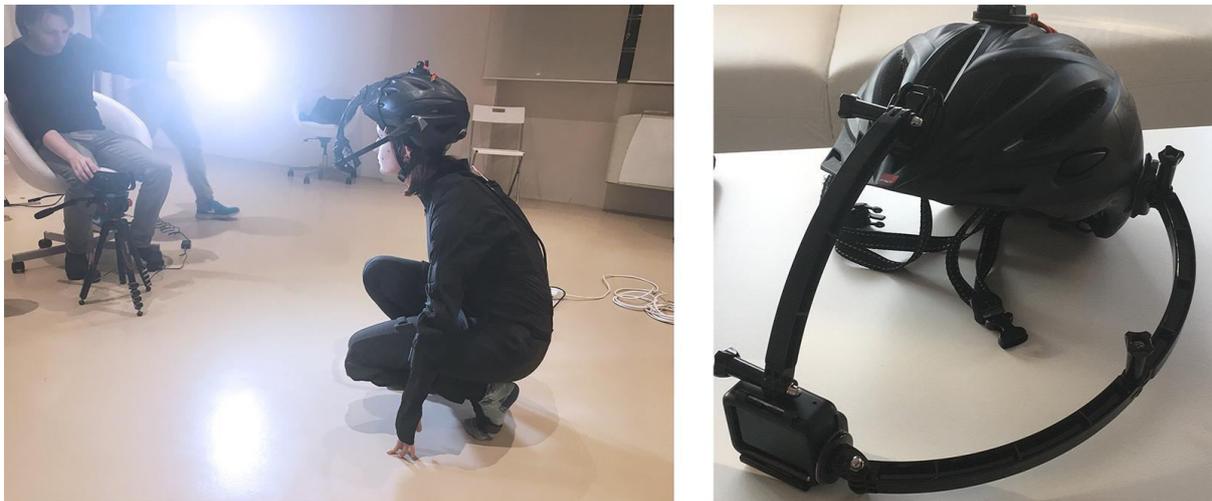


Figura 7.1: Elmetto per motion capture facciale creato in studio.

7.1 Faceware Analyzer

Analyzer è il software destinato al tracking facciale, e uno dei punti chiave del programma è che permette di effettuare un tracking selettivo sulle aree del viso, diversamente dagli altri tool in commercio che invece svolgono un lavoro di tracking globale. Durante l'installazione viene richiesto prima di tutto di installare Matlab Runtime Compiler (MCR), un set autonomo di librerie condivise che consente l'esecuzione di applicazioni compilate o componenti da Matlab su computer dove non è installato. Il carico di lavoro viene suddiviso tra quattro macro-aree del volto: gli occhi, il naso, le sopracciglia e le labbra e questa suddivisione permette uno studio approfondito e preciso dei punti di tracking. Il punto in comune fra le componenti sono le narici, che servono per stabilire la posizione relativa fra le componenti in esame infatti, una volta terminato il tracking di una componente si bloccano i punti di tracking relativi al naso nelle posizioni già registrate, in modo da restare fisse per le altre e garantire una coerenza globale del sistema.

A inizio della sessione, viene richiesto di creare un nuovo *job* nel quale viene caricata la clip video registrata precedentemente, e impostare alcuni parametri come la lunghezza della sorgente. L'interfaccia utente è molto semplice e intuitiva: un pannello a comparsa permette di selezionare la regione del viso su cui eseguire il tracking, mentre quella complessiva 'face' è da utilizzarsi come ultimo controllo prima dell'esportazione. Si distinguono due fasi di lavoro:

- 1) Training - letteralmente si insegna al programma a riconoscere i connotati del viso. Ne servono almeno due per effettuare un tracking, ma è caldamente consigliato di inserirne di molteplici, soprattutto per salvare alcune pose chiave. Nel caso particolare delle labbra bisogna prestare più particolare attenzione: è consigliato inserire una posa di training per ogni espressione differente, ma di non aggiungere troppe chiavi simili altrimenti si rischia di mandare in confusione l'algoritmo che procede al tracking. Il processo di training si effettua diverse volte finchè non si ottiene un tracciamento uniforme e i tracker non vengono mai persi.
- 2) Tracking - il software elabora i keyframe del training interpolando le varie pose, la precisione dipende dal numero di frame di training. Il calcolo viene effettuato separatamente per ogni sezione del viso e può succedere che a volte il tracking venga perso, ma è possibile andare a sistemare aggiungendo dei keyframe di posa oppure con il tasto CANC, che applica un'interpolazione tra il frame precedente e quello successivo. L'ultima soluzione si presta decisamente bene per quando si perde un singolo frame o un range piccolo che conviene ripristinare a mano.

Approvato il tracking si può procedere con l'esportazione, sotto la voce parametrizzazione, un processo che trasforma i dati in maniera che possano essere utilizzati all'interno del programma compagno, Faceware Retargeting. Indicativamente si crea una mappatura di coordinate spaziali che serviranno per pilotare i controllori del rig del personaggio, e quindi animarlo.

7.2 Faceware Retargeting

È il programma di riferimento per svolgere il ruolo del face retargeting, ed esistono diverse versioni plugin che si interfacciano con i principali software di manipolazione del 3D, incluso Maya. Anche in questo caso è necessario *insegnare* al programma come riconoscere un

modello facciale, e il primo passo da compiere per arrivare ad animare è quello di impostare il viso del personaggio. Inizialmente viene richiesto di indicare tutti i controllori per le deformazioni del viso, siano essi driver o semplici ossa con controller FK: si crea un nuovo sistema di *Pose Groups* che è suddiviso esattamente come le sezioni utilizzate da Analyzer (bocca, sopracciglia, occhi). Nel caso si voglia utilizzare la funzione di autosolve, un'interfaccia grafica guida l'utente attraverso tutte le pose chiave che devono essere registrate come punto di riferimento per quando si effettuerà il retargeting, cercando di pre-comporre una libreria di espressioni che verrà utilizzata per interpolare le pose del personaggio. L'autosolving non è compatibile con il retargeting manuale e viene più volte suggerito di scegliere a priori la strada da seguire. La funzione automatica funziona bene per tutti quei personaggi 3D che compaiono magari come personaggi di sfondo, oppure nei casi in cui non è richiesto un alto grado di realismo nella resa. Per questi motivi si è scelta la strada manuale, per poter estrarre la miglior performance possibile da utilizzare nel teaser trailer prodotto durante la tesi, dove verrà data voce alla protagonista tramite un monologo che fisserà lo sguardo sul suo primo piano, quindi condizione necessaria per un buon risultato è sicuramente quello di ottenere delle ottime espressioni facciali. Nella barra *performance* si può procedere a caricare il file .fwr generato da Analyzer e il corrispettivo video sorgente da usare come reference, comprensivo di audio. Da questo punto inizia un lavoro di comunicazione con il programma per animare il volto, un compito che terminerà una volta soddisfatti dei risultati.

7.3 Pipeline per il Motion Capture Facciale

L'immagine sottostante rappresenta graficamente la pipeline per il facial motion capture adottata per la fase sperimentale di questa ricerca, visualizzando il processo del caso studio appena esaminato e come i vari strumenti si interfacciano tra di loro.

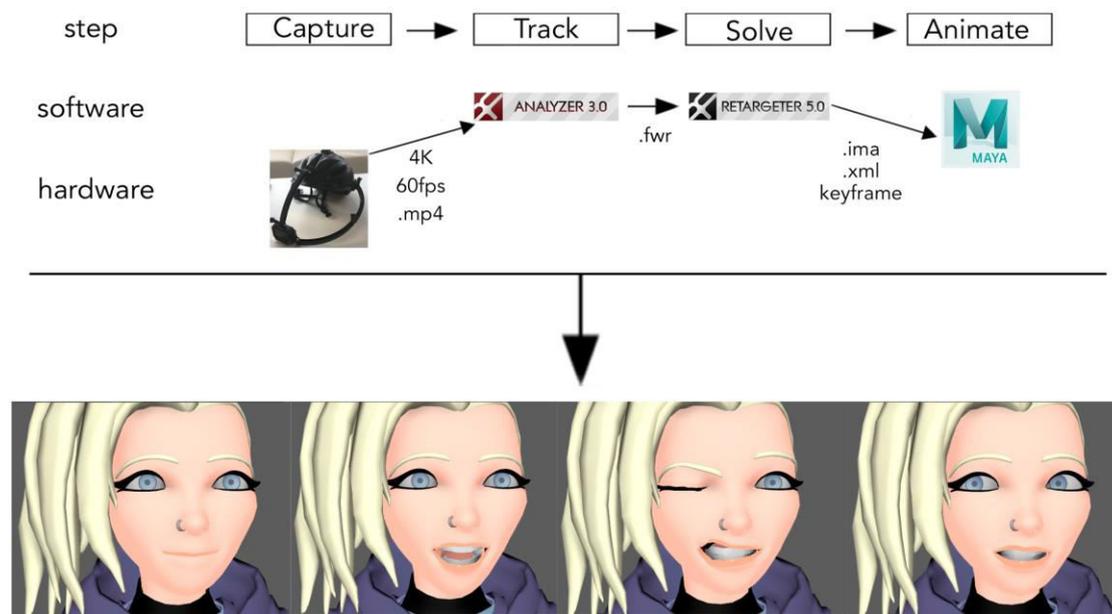


Figura 7.2: Flusso di lavoro per il motion capture facciale, comprensivo degli strumenti e dei programmi utilizzati per ottenere l'output finale.

Parte II

Produzione Indipendente

La fase sperimentale che mette in gioco tutte le conoscenze acquisite durante la stesura dell'elaborato è trattata esaustivamente in questa sezione. Il progetto Reverie Dawnfall è una serie d'animazione iniziata insieme a questo progetto di tesi, e raccoglie tutti gli sforzi finora compiuti per raggiungere una produzione di alto livello con le tecnologie più all'avanguardia accessibili per il mercato indipendente. L'obiettivo raggiunto in questa parte di elaborato è la presentazione della prima versione del teaser trailer del progetto, che sarà mostrato in anteprima durante la discussione di questa tesi, alla fine del percorso di studi.

8. Progetto Reverie Dawnfall

Il progetto Reverie Dawnfall nasce come idea per una serie tv animata targettizzata ambientata in un futuro distopico, dove l'umanità è ridotta all'osso da guerre biochimiche e dalle condizioni ambientali ormai estreme. In questa ambientazione si mescolano e si inseriscono alla perfezione elementi futuristici e cyberpunk, caratterizzando tutti i personaggi in maniera unica. L'idea trova le sue radici nei classici della cultura pop giapponese contemporanea, come *Il Mistero della Pietra Azzurra* (1991), *Ergo Proxy* (2006), *Battle Angel Alita* (1991) e altri interessanti titoli di stampo più occidentale come *Saga* (2012), ed è stata sviluppata per un target teen-adult. All'interno della storia si svolgono diverse dinamiche che spingono a una profonda riflessione su diverse tematiche: l'ecosistema al collasso, il genere umano pesantemente mutato dalle radiazioni lasciate dalla guerra e il desiderio innato di riscoprire il passato sono solo alcuni pezzi di una storia che promette di essere avvincente e adatta per un pubblico maturo.

Attualmente l'arco narrativo è composto da tre stagioni di una decina di episodi ciascuno, e vedono come protagonista una giovane e brillante studentessa di entomologia di nome Nadya Sinkamen. L'avventura inizia in una città protetta da una cupola gigante, mentre fuori l'aria è tossica e il mare abitato da un tappeto di meduse mortalmente velenose; la

popolazione è alle prese con continue rivolte popolari sotto l'influenza di cinque grandi superpotenze che si riflettono in multinazionali dal controllo monopolistico. In una società dove ogni essere umano soffre di qualche malattia o alterazione genetica, la protagonista crede di soffrire di sinestesia, una condizione che la porta ad avere visioni di un mondo simile a quello che conosce ma vivo e luminoso: nel momento in cui queste visioni si fanno più nitide e insistenti, con attacchi violenti che la dissociano completamente dalla realtà, Nadya inizierà a dubitare del proprio universo, e con l'aiuto dei suoi amici intraprenderà un viaggio pieno di pericoli e imprevisti alla ricerca di un mezzo per salvare il loro mondo.

Il progetto nasce in un ambiente totalmente indipendente, di fatto al momento è auto-prodotto dallo sforzo e dall'ambizione di Riccardo Antonino, professore di Effetti Speciali al Politecnico di Torino e uno dei fondatori di Robin Studio, presso il quale si svolge la produzione. Questa serie nasce quindi dal forte desiderio di realizzare un prodotto multimediale animato fortemente competitivo nel panorama televisivo nazionale, con l'aspirazione di espandersi anche al di fuori del confine del nostro Paese. Lavorare in uno scenario *indie* pone diverse sfide, di carattere economico e tecnico, che solo con un team affiatato e totalmente fiducioso nella riuscita del progetto si è in grado di superare. Per questa ragione ci siamo inizialmente dedicati alla formazione di un gruppo di lavoro affidabile e competente, ad oggi composto da:

- Riccardo Antonino - ideatore, supervisore e produttore;
- Melissa Coarezza - responsabile reparto animazione e rigging, supervisore;
- Mark Gore - sceneggiatore e supervisore;
- Edoardo Audino - designer dei personaggi e storyboard artist;
- Michele Cannata - responsabile reparto modellazione;
- Maddalena Negrini - voce e attrice di motion capture per Nadya.

La produzione di tutto il progetto è iniziata all'inizio di Marzo, poche settimane dopo l'inizio di questo elaborato finale che intendeva appunto trovare uno scopo pratico e creativo allo studio della tecnologia di motion capture. Fino adesso, il calendario di produzione si può riassumere nella seguente immagine:

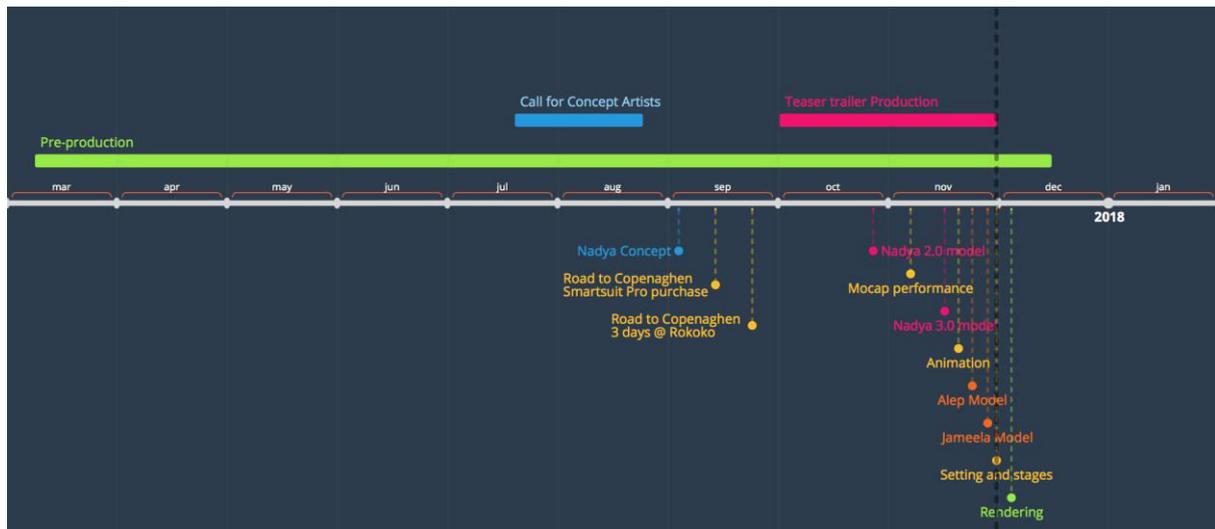


Figura 8.1: Timeline del primo anno di produzione

8.1 Pre Produzione

Nei primi mesi ho pianificato diversi colloqui con Riccardo Antonino con l'obiettivo di iniziare una stesura del progetto, prendendo nota dei punti salienti della trama e iniziando a tracciare dei profili per i personaggi principali. Nel mese di giugno è entrato ufficialmente nel team di produzione Mark Gore come principale sceneggiatore degli episodi: grazie a lui il progetto ha iniziato a concretizzarsi in una scaletta di episodi più definiti, e lavorando a stretto contatto con Riccardo è riuscito a stravolgere in meglio quella che era l'idea primordiale della storia e a renderla più avvincente. I personaggi venivano via via più definiti e iniziava a crescere l'esigenza di avere in squadra un character designer che si occupasse di dar vita alle parole descrittive nelle schede personaggio, che fino a quel momento si basavano su tavole di reference studiate grazie alla fonte inesauribile di immagini e ispirazione che è Pinterest.

Verso fine luglio abbiamo aperto una call ufficiale per un character designer che curasse tutte le bozze dei personaggi e la risposta è stata ampiamente positiva sicché abbiamo avuto modo di conoscere sei artisti, ognuno con un proprio stile caratterizzante. La scelta è ricaduta su Edoardo Audino, uno studente al secondo anno dell'Accademia Albertina, perché il suo stile riflette esattamente l'idea grafica che tutti e tre avevamo in mente. Inoltre eravamo tutti d'accordo sul fatto che i suoi disegni potessero essere facilmente riutilizzabili in caso di un adattamento fumettistico del progetto. Da inizio settembre è quindi Edoardo a occuparsi delle bozze di tutti i personaggi, dalle moodboard alle t-pose per la modellazione e il suo

metodo di lavoro è molto veloce e questo permette di parallelizzare il processo e di scambiarsi feedback durante le varie fasi di sviluppo creativo.

Un altro enorme aspetto da considerare nel lavorare in questo ambito è che per molti elementi del gruppo si tratta della prima vera esperienza a livello professionale di questo tipo, quindi è positivo che ci sia un'attitudine positiva nel ricevere consigli e suggerimenti per migliorare le proprie capacità.

Con la prima stagione interamente abbozzata ci siamo interrogati sulla metodologia migliore da seguire per la produzione, e si è deciso di voler creare un teaser trailer che presentasse la protagonista e la storyline generale da distribuire in rete e durante eventi mirati per iniziare ad attrarre potenziali investitori. Sempre nel mese di settembre era previsto il viaggio in Danimarca per un training accelerato sulla tuta di Motion Capture che era stata appena acquistata, e per l'occasione ho modellato una prima versione di Nadya utilizzando il software Autodesk Maya come strumento designato all'intera produzione, approfittando della licenza studenti disponibile gratuitamente. Nel frattempo si è aperta un'altra call per un character artist che si occupasse dei modelli 3D dei personaggi, un altro traguardo necessario per poter impostare un workflow di lavoro settorializzato e specializzato. A inizio ottobre si è aggiunto quindi un nuovo tesista al progetto, Michele Cannata, che come oggetto della sua tesi ha proprio quello di studiare una pipeline di modellazione di personaggi per una produzione indipendente.

A ottobre è ufficialmente iniziata la produzione del teaser con il team al completo, argomento che sarà trattato in dettaglio nei paragrafi successivi.

8.2 Personaggi

Di seguito si vogliono presentare brevemente alcuni dei personaggi principali che animeranno le avventure del progetto Reverie Dawnfall, descrivendo il loro background e caratteristiche note e mostrandone il profilo grafico così come disegnato da Edoardo. In particolare modo, i primi personaggi descritti sono quelli selezionati per il teaser trailer, e che quindi verranno modellati e animati.

8.2.1 Nadya Sinkamen

Figlia di due dipendenti della Pharmacopia (una delle cinque superpotenze) residenti nella seconda città più grande del Paese, Nadya ha 19 anni e studia entomologia comportamentale all'università di Dome City. Presenta un carattere curioso e tenace, velato da un cinico distacco accompagnato da una nota di sarcasmo, ed è profondamente legata ai suoi amici. Soffre di sinestesia, una condizione che la porta ad avere vere e proprie allucinazioni psichedeliche: per questo possiede alla base della nuca un impianto biomeccanico che le consente di iniettare la sua dose giornaliera di farmaco e tenere a bada il suo male, di cui le prova una forte paura.

E' sempre accompagnata dal suo fedele compagno a sei zampe Alep, un insetto simile a uno scarabeo ma alto circa venti centimetri e in grado di camminare in posizione semi-eretta che ha creato lei.

Nadya è il personaggio su cui si è maggiormente lavorato fino ad oggi, non solo perché si tratta della protagonista ma perché in quanto tale è la prima a comparire sullo schermo, quindi tutti gli esperimenti grafici e di animazione sono stati effettuati su di lei, in modo da stabilire un processo di creazione uniforme per tutti i personaggi a seguire.

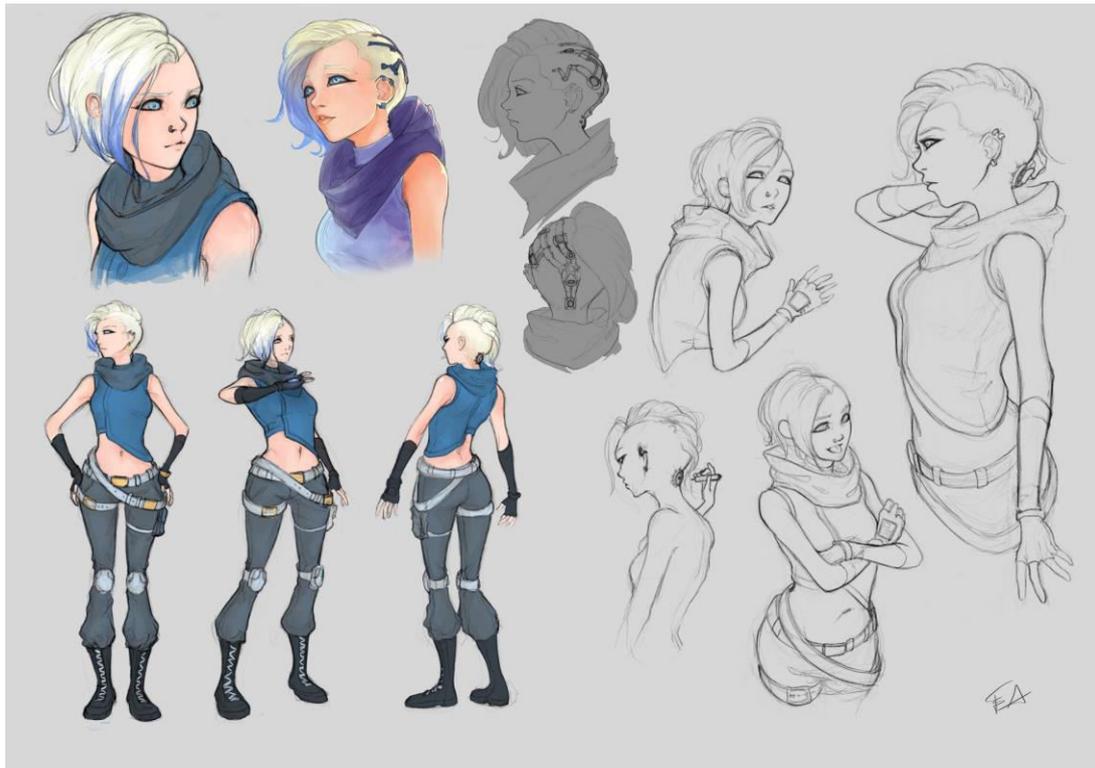


Figura 8.2: Design di Nadya (in alto) e una prima versione renderizzata in 3D (in basso)

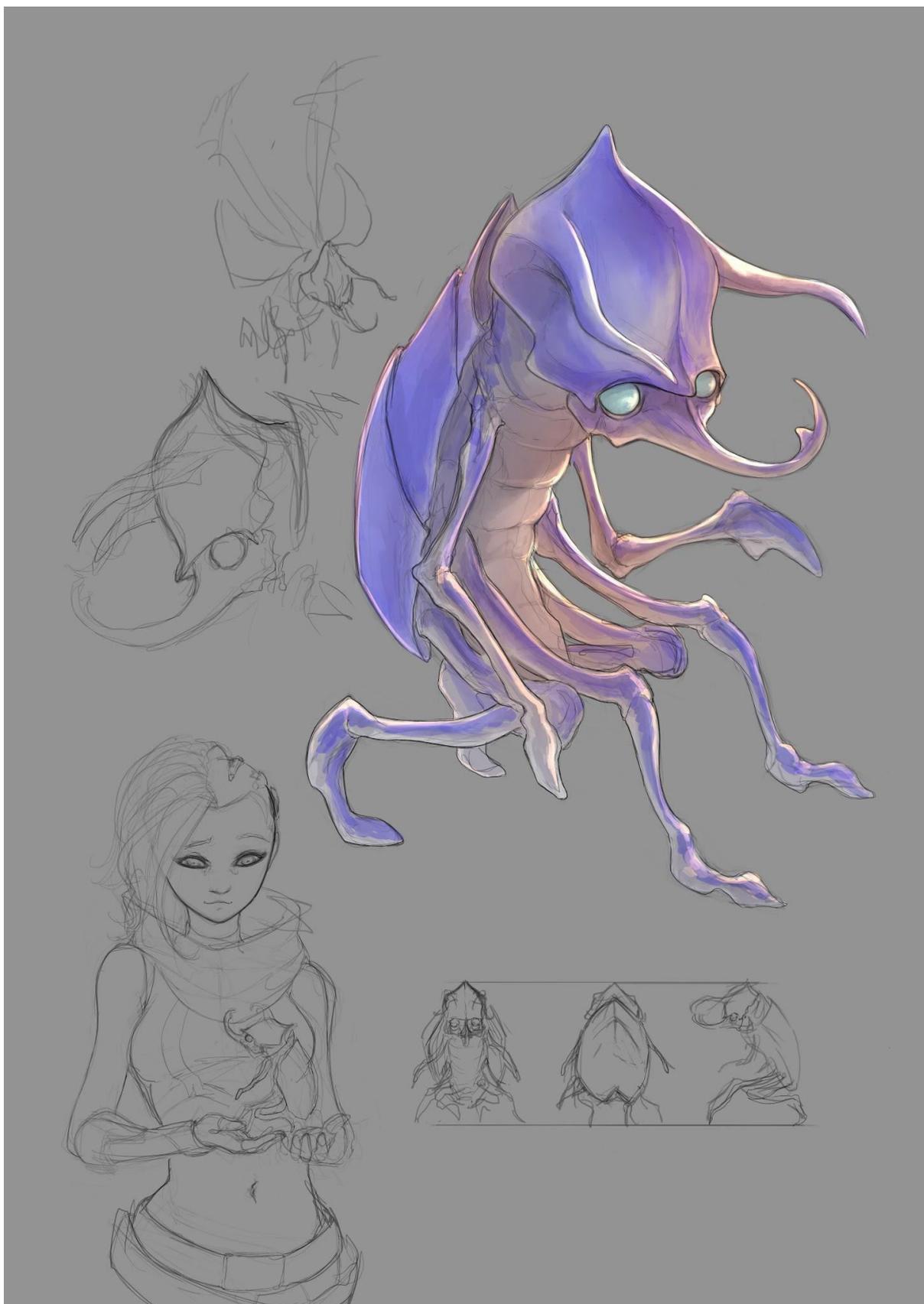


Figura 8.3: Design di Alep, e studio delle proporzioni rispetto a Nadya

8.2.2 Jameela Rani

È un'amica di Nadya di un anno più giovane e sua compagna di stanza all'università, ha un carattere molto pragmatico e intuitivo, seppur mostri un'indole rassegnata e disfattista che rasenta un certo nichilismo. Nata con un solo arto sano, quello del braccio sinistro, Jameela ha origini molto umili, e non potendosi permettere delle vere protesi cyborg si costruisce gambe e braccia con tutte le componenti meccaniche che riesce a riciclare: parti robotiche, elettrodomestici, circuiti, fanno tutti parte del suo corredo che subisce continui cambiamenti a causa dei materiali di fortuna con il quale vengono assemblati. In confronto a una Nadya dalla pelle quasi diafana, lei ha la carnagione abbronzata sulla quale risaltano i suoi occhi verdi e i capelli castano rossiccio.

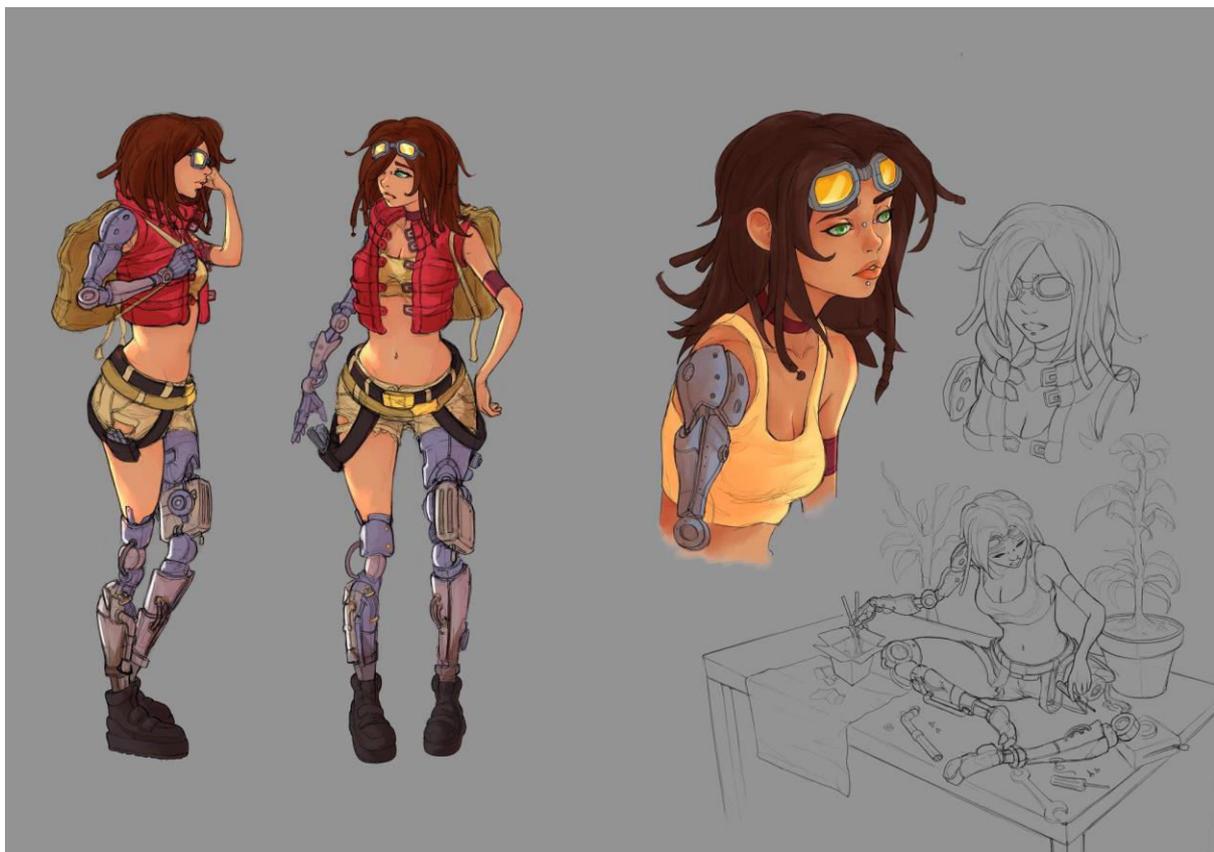


Figura 8.3: Character design di Jameela

8.2.3 Peregrine

E' una delle personalità più interessanti e complesse che caratterizzano l'universo narrativo: a differenza del resto del genere umano, Peregrine è l'unico individuo di cui sia nota l'esistenza a essere nato senza alcuna patologia o difetto genetico. Da giovane decise di sfruttare questo suo 'dono' per fare soldi e firmò con la Pharmacopia un contratto per essere studiato al fine di sintetizzare un farmaco basato sul suo DNA, chiamato Theriac, ma gli esperimenti effettuati su di lui gli hanno causato gravi danni ai centri sensoriali e alle terminazioni nervose. Senza l'impianto sulla sua testa sarebbe completamente sordo e senza alcun senso dell'equilibrio, un occhio è ormai completamente cieco e appare inquietante e lattiginoso, e non prova più alcun tipo di emozione.

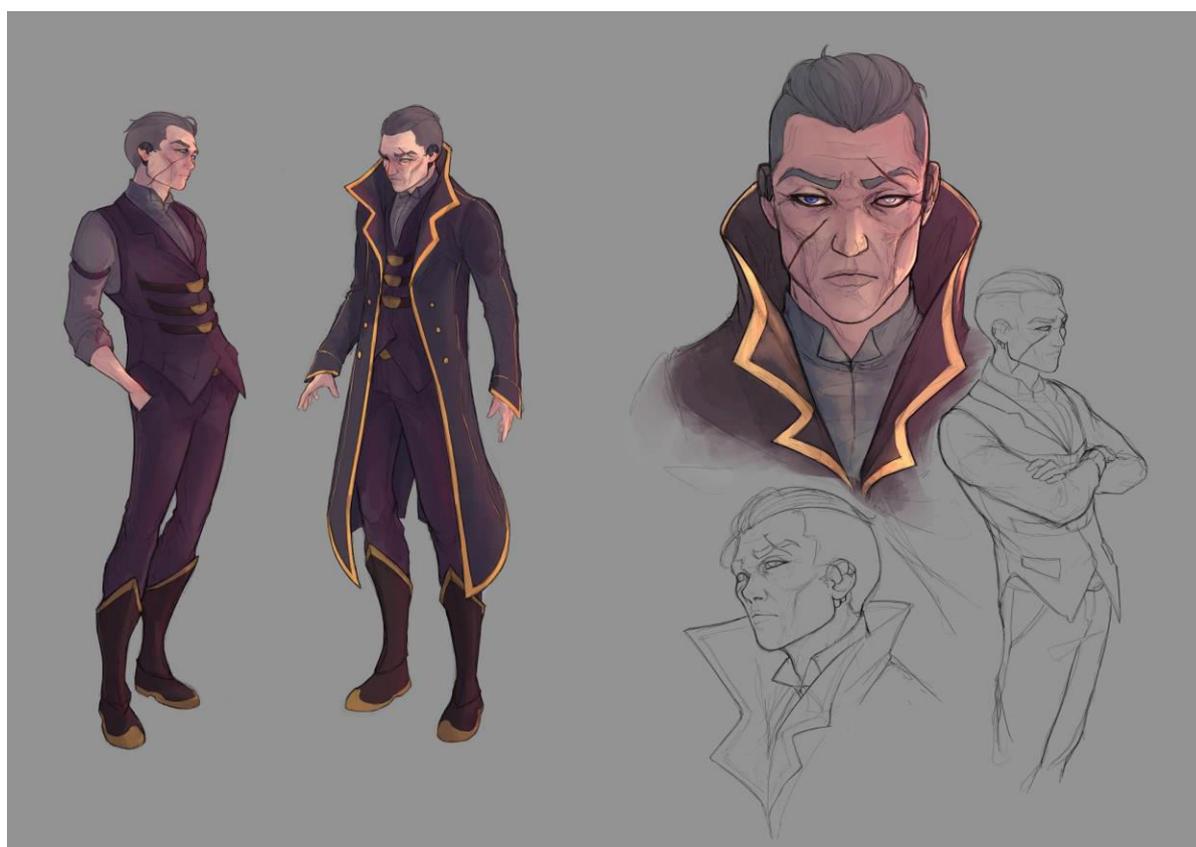


Figura 8.4: Character design di Peregrine

Il suo corpo è coperto da cicatrici, segno indelebile degli orrori vissuti, alcune se l'è procurate lui stesso alla ricerca di qualsiasi tipo di sensazione. Peregrine ha 68 anni e fisicamente ne dimostra la metà, grazie al gene che lo rende praticamente invincibile a ogni agente patogeno e all'avanzare dell'età, ma la sua vita è un susseguirsi di tragedie: moglie e figlio sono morti a causa di patologie che non erano ancora curabili perché il Theriac non era ancora sul mercato.

A causa di tutti questi eventi Peregrine è nel presente un uomo spietato e inclemente, incapace di provare emozioni eppure ossessionato dal voler trovare una cura al suo apatismo. La conoscenza con Nadya sarà uno dei principali motori della vicenda narrativa.

Può essere considerato come il villain del primo arco narrativo, anche se le sue sfortune lo rendono un personaggio fortemente empatizzato. Con il susseguirsi della storia si scoprirà un lato più umano di questo personaggio che lo condurrà attraverso un cammino di redenzione e pacificazione.

8.2.4 Breather

Un altro personaggio particolare dell'universo Reverie Dawnfall e amico di Jameela e Nadya, Breather è praticamente la coscienza del gruppo. Nel suo caso, la malformazione con cui è nato lo costringe a vivere con un respiratore poiché i suoi polmoni necessitano di aria ad elevatissimo tasso di umidità. Per di più presenta sul collo delle branchie che in realtà non sono funzionali al suo sistema respiratorio, quindi praticamente solo estetiche e inutili. Il respiratore è sempre sulle spalle grazie a una bombola a zaino, e la mascherina si posa sulla bocca e sul naso, coprendo in parte il viso pallido. E' curioso come solo all'interno della serra in cui lavora e si trattiene spesso la notte sia in grado di respirare senza la mascherina, in quanto l'umidità di quel luogo gli garantisce un benessere che non è in grado di trovare altrove a causa dei cataclismi causati dal Regime. Rispetto a Jameela è decisamente più positivo come carattere, razionale ma solare e ironico. Il suo nome di battesimo è Ritvars ma è comunemente chiamato Breather per il suo marchingegno vitale.

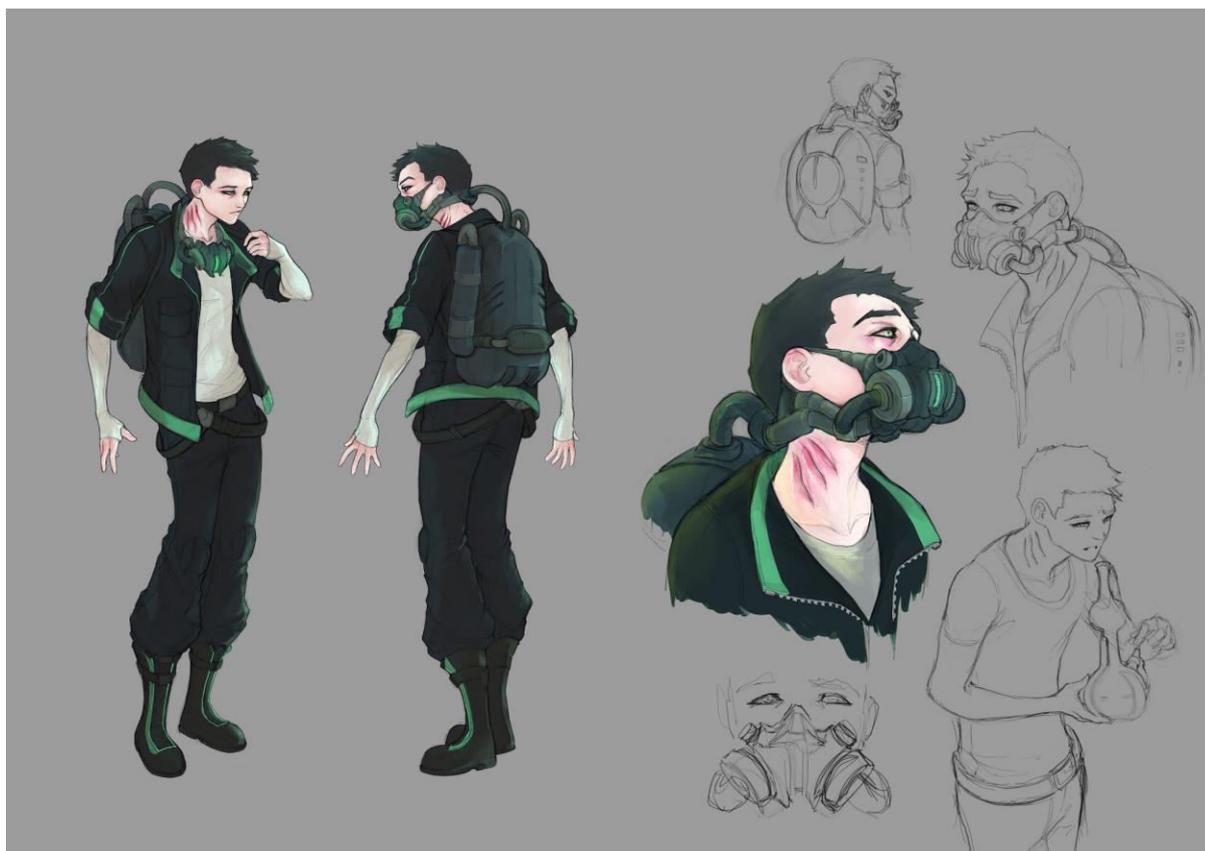


Figura 8.5: Character design di Breather

8.3 Ambientazione

Reverie Dawnfall è ambientato su un esopianeta simile alla Terra come morfologia e caratteristiche, con una differenza sostanziale che si riflette nell'assenza del moto di rotazione, un elemento che rende perennemente notturna una metà del pianeta, e perennemente giornaliera l'altra metà. La linea di confine delle due facce vive in un crepuscolo eterno, e questo particolare momento della giornata scandisce i colori e lo stile dell'illuminazione globale dell'intera serie animata.

La città dove si svolge la prima stagione si trova sul margine estremo della zona crepuscolare, e sorge su una baia denominata *Poisoned Seas* poiché abitata da meduse velenosissime. La cupola a protezione della città serve a preservare l'atmosfera vivibile in un contesto altamente inquinato e tossico in cui si trova il modo esterno: la conformazione della città quindi si erge secondo la semisfera che la protegge, con gli edifici più alti verso il centro e via via sempre più bassi verso i confini della cupola. La zona notturna si trova ad est e la città si trova proprio sul confine, quasi a sfiorare quella zona.

Dome City è demograficamente la quarta città più grande della nazione, e ospita al suo interno la più antica università del pianeta poichè la città era stata strategicamente costruita sul confine con l'eterna notte per studiarne la flora e la fauna. Grazie alla sua ubicazione e al suo polo culturale, è uno dei centri economici di riferimento per il pianeta. L'architettura che si presenta è mista, distribuite radialmente all'interno della cupola: gli anelli più esterni ricordano una metropoli asiatica dei giorni nostri, mentre avvicinandosi al centro si trovano i grattacieli più futuristici e irregolari.

L'ambiente esterno circostante è simile a una tundra fatta di basse colline erbose e con poche conifere sparse così come lo sono i laghi. La vegetazione è di svariati colori porporini e muschiosi, e ci sono dei laghi acidi che ricordano molto quelli del parco di Yellowstone.

Tutti questi elementi sono posti in modo da portare sempre lo spettatore a chiedersi se si stia trovando effettivamente sul nostro pianeta e quali siano le cause che l'hanno portato a mostrarsi come lo vede dipinto.



Figura 8.6: Environment design di Dome City vista dall'esterno (© Federico Ferrarese)

9. Teaser Trailer

Per presentare brevemente e in maniera accattivante il progetto abbiamo deciso di far parlare la protagonista assoluta della storia, Nadya. Abbiamo ipotizzato uno scenario di apparente normalità per spiegare tutto quello che succede nella sua vita e di un mistero in arrivo. Il format di questo prodotto è quindi un teaser di circa due minuti dove la protagonista racconta sotto forma di v-log la vita a Dome City. Il suo monologo è riportato alla fine del capitolo. Si è scelto di far recitare Maddalena in lingua inglese per poter avere la possibilità di mostrare il prodotto anche all'estero e aumentare quindi la visibilità su scala internazionale.

L'ambiente circostante è la camera da letto di Nadya e Jameela, una stanza di un dormitorio universitario piena di circuiti e altre cianfrusaglie con il quale Jameela sperimenta e costruisce le sue protesi. Grazie a questa scelta di produzione è possibile visualizzare facilmente alcuni elementi chiave del progetto: una parte dell'ambientazione, futuristica con elementi cyberpunk, e alcuni dei protagonisti e personaggi più particolari che animano il mondo. Nella scena rappresentata nulla è mostrato per caso, e nessuna parola è lasciata al caso, è possibile visualizzare tutti gli elementi che faranno da sfondo alle vicende dei protagonisti.

Con l'inizio della produzione del trailer si sono verificate le prime rivisitazioni dei personaggi e dello stile grafico che si voleva ottenere, è stato quindi necessario trovare una pipeline che permettesse di lavorare in maniera congiunta, in modo da agevolare le modifiche in corso d'opera sia nella trasformazione del personaggio 3D, e di conseguenza nella fase di nuova animazione. Gli strumenti e i software utilizzati per la realizzazione del trailer sono molteplici e verranno nominati più volte nel corso dei prossimi paragrafi, se ne offre un elenco sintetico per spiegarne brevemente il ruolo:

- *Smartsuit Pro* per il motion capture del corpo;
- *GoPro* ed elmetto costruito per il motion capture del viso;
- *Maya* come ambiente di lavoro durante la pipeline 3D;
- *Advanced Skeleton*, uno script MEL usato per il rigging del viso in Maya;
- *Faceware Analyzer* e *Retargeter* per il tracking e il retargeting facciale;
- *Adobe After Effects* per il compositing.

9.1 Sceneggiatura

NADYA

"Ok, so... name is Nadya. With a y.

Think that's weird? Tip of the iceberg.

Not that there are anymore. Icebergs, that is. Where was I?

Right: weirdness.

So... I see sound. I can taste music and feel colors with my skin.

See? I told you it would get weird.

It's called synesthesia and it makes my life kind of hectic.

Well, that's also because we live under a totalitarian, decadent regime after the meltdown of the planetary ecosystem.

Oh yeah, the world is largely unlivable: resources are scarce, the seas are poisoned

and the air toxic. So much so that people are born either crazy or deformed of both.

Which is why we live under huge domes protecting our cities.

Kind of a bummer, I'm an entomologist and all the most intriguing bugs are outside.

As is the answer to our mystery here.

Because there is a mystery... didn't I mention the mystery? There's one.

But I'm not telling! Not right away.

I'm not afraid of anything, certainly not of you getting mad at me, so go ahead, call

this what it is: a damn teaser."

10. Pipeline di Produzione per il teaser

La fase di creazione del personaggio è stata reiterata più volte per venire incontro alle esigenze stilistiche della produzione. Il modello di partenza di Nadya si è rivelato inadatto per le sue forme troppo rotonde, rispetto allo stile grafico che si voleva raggiungere. C'è stato uno studio stilistico fin dalle fasi preliminari del progetto, volgendo lo sguardo ai lavori dell'artista irlandese David O'Reilly e del successo tutto italiano di *Gatta Cenerentola* (2017). L'elemento comune dei due lavori che si vuole mantenere è uno stile poligonale non troppo complesso, e in assoluto lo stile dell'illuminazione.

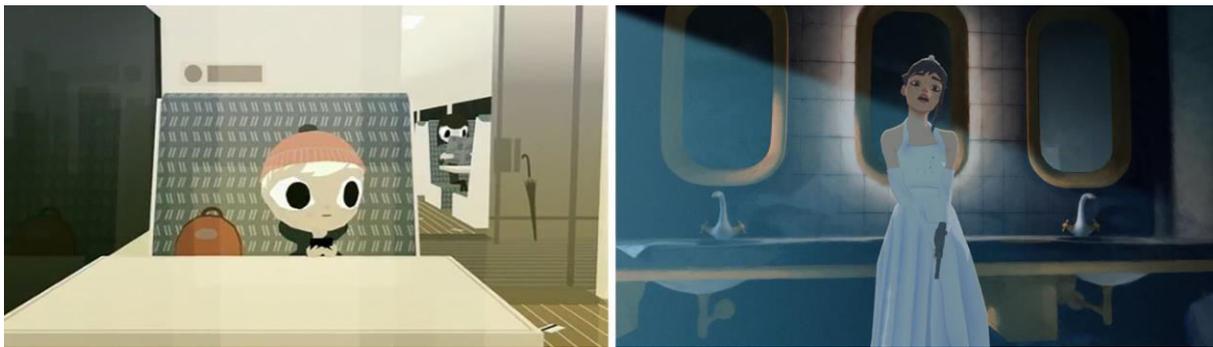


Figura 10.1: U2, *'I'll Go Crazy if you don't go crazy tonight'*, videoclip animato da David O'Reilly (a sinistra).
Mad Entertainment, *Gatta Cenerentola* (a destra)

Il primo compito ha quindi riguardato la modifica del modello 3D di Nadya perché rispondesse a due requisiti, uno riguardante il numero di poligoni e l'altro che sfruttasse la spigolosità data dai pochi poligoni come fattore artistico. Il modello iniziale mostrato nel paragrafo 8.2.1 presenta circa 38k vertici e l'obiettivo è stato quello di dimezzare almeno tale valore, uno scopo che è stato raggiunto tramite tecniche di retopology con il software 3D Coat studiate da Michele nel corso della sua tesi. Il modello retopologizzato di Nadya usato per il trailer prodotto per questa ricerca è composto da 17k vertici, e non è ancora il modello definitivo in quanto si stanno studiando tecniche per diminuire ulteriormente il peso del modello da utilizzare nella produzione vera e propria di scene complesse animate da diversi personaggi e ambienti più dettagliati. Tutte queste accortezze sono d'obbligo in un ambiente dove le macchine destinate al calcolo del rendering non sono di fascia alta e non si dispone di render farm apposite per quel compito: più la scena è densa di poligoni e complessa nell'illuminazione e più il rendering richiederà potenza di calcolo, un tempo che si è cercato

di ridurre al minimo durante la produzione del trailer grazie a uno studio mirato sulla modellazione.

Parallelamente alla modellazione sono avvenute le sessioni di performance capture con Maddalena e i primi studi sul rigging per trovare la metodologia più efficace e semplice da applicare a più personaggi.

Finalizzato il modello, ho potuto mettere in pratica gli studi effettuati sul rigging ottimizzato per il Motion Capture che si sono compiuti con le versioni precedenti di Nadya. Mentre si procedeva con lo studio di tecniche per il texturing del personaggio, ho iniziato a ripulire i dati dell'animazione mocap direttamente sullo scheletro Rokoko, ripetendo alcuni dei procedimenti chiave studiati nelle prime fasi di analisi della pipeline, come descritti nel caso studio di Rokoko.

Si sono studiati diversi approcci per il rigging del viso e del corpo, via via migliorati grazie alla ripetizione dei passaggi ogni volta che veniva consegnata una versione aggiornata del modello di Nadya: sfruttando l'occasione per affinare la tecnica, il rigging è stato il primo strumento di valutazione della bontà del modello 3D, se la sua topologia fosse corretta e funzionale per l'animazione. In questo modo si è potuto lavorare insieme fino ad ottenere un modello ideale, tale da essere utilizzato per il trailer.

Nella pulizia dei dati mocap ho effettuato le prime prove con gli animation layers come strumento risolutivo di correzione per quei dati corrotti che si sono trovati anche nel primo esperimento a Copenaghen, dove l'intervento manuale non era sufficiente a garantire un'animazione fluida.

Terminata e ufficializzata l'animazione, si è lavorato intensamente sulle ricerche per il rendering, e nonostante il poco tempo rimasto a disposizione si è riusciti a creare uno stile che iniziasse ad accostarsi al livello di dettaglio di colore desiderato.

La pipeline è stata formata da processi spesso ricorsivi in quanto le modifiche di progetto hanno conseguentemente richiesto di tornare indietro su alcuni passaggi già assodati, ad esempio il cambio dei capelli del modello di Nadya ha fortemente rallentato i lavori nell'ultimo periodo di produzione del teaser perché ha costretto il reparto tecnico a tornare indietro alla modellazione, e ripetere in sequenza tutti i passi che seguono dopo.

Di seguito si vuole esporre il resoconto tecnico dettagliato di questi passi che hanno costituito il lavoro principale svolto autonomamente per questa attività di ricerca.

10.1 Retopology

Prima di affidare completamente la pipeline di modellazione al collega Michele Cannata, in funzione della richiesta di diminuire il numero di vertici ho effettuato una prima revisione del nuovo modello di Nadya che era già stato modellato. Si è applicato un metodo manuale e selettivo per la rimozione degli *edge loops* laddove non erano previste deformazioni particolari, lasciando maggiore densità di poligoni attorno alle aree adiacenti alla bocca, agli occhi e nelle ginocchia e nei gomiti. Nell'esempio riportato in Figura 10.2 si confrontano le due teste: quella di sinistra conta 2300 vertici ed è stata la base su cui ho effettuato il retopology, arrivando al modello di destra che presentava 1450 vertici. Questo tipo di procedura è utile in fase di valutazione dei limiti per il retopology del modello ma non è ideale per l'utilizzo in pipeline di produzione perché si è fatto uso di un approccio aggressivo e non conservativo della morfologia del personaggio. Questo studio è stato seguito e successivamente eseguito dal mio collega, che ha usato questa prova come base per conoscere i punti necessari e sufficienti per poter avere delle animazioni realistiche con il numero più basso possibile di vertici.

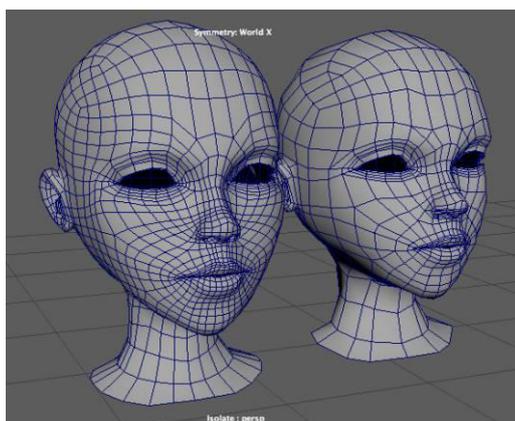


Figura 10.2 (a): Retopology del viso di Nadya, a sinistra 2300 vertici, a destra 1450.



Figura 10.2 (b): Retopology del modello 2.0 di Nadya (a destra in figura). Il modello retopologizzato manualmente (a sinistra in figura) conta circa la metà dei vertici, da 38k a 19k. Il nuovo modello mostra gli studi iniziali sugli shader.

10.2 Rigging

Un qualsiasi personaggio digitale per essere animato deve prima essere configurato: il processo di *character setup*, o rigging, consente di definire come il modello sarà animato tramite la creazione di vincoli e deformati appropriati. Il rigging non si limita solo ai personaggi ma anche agli oggetti che richiedono particolari deformazioni (es. Indumenti e oggetti inanimati) e la prima regola di questo tipo di lavoro è che non ci sono limiti nella definizione di un rig, se non quelle dettate dall'esigenza di produzione.

Riggare un personaggio significa creare uno scheletro, o un'armatura, al quale la mesh viene associata per essere animata secondo alcune regole di impostazione, date da vincoli di deformazione e parenting tra le ossa. Gli scheletri sono composti da gerarchie di ossa che sono suddivise in famiglie di giunti e si diramano tutti da uno giunto radicale che nei personaggi è tipicamente situato nei fianchi: ad esempio, il giunto del ginocchio è genitore del giunto della caviglia ed è figlio del giunto del fianco.

Il passaggio che permette allo scheletro di deformare la mesh del personaggio è definito *skinning*: per ogni osso si definisce quale porzione della mesh viene deformata, e in quale percentuale di influenza, più ossa possono avere controllo sulla stessa area di poligoni, con peso diverso. Per definire il contributo di ogni giunto alla deformazione dei vertici di una mesh si possono usare diversi metodi come il *weight painting* oppure con la selezione

manuale dei vertici; il primo metodo è più raffinato e permette di ottenere dei risultati più professionali in quanto permette di andare a definire con precisione la percentuale di influenza delle varie ossa. Con lo skinning, il personaggio è legato allo scheletro ed è potenzialmente pronto per essere messo in posa.

Le ossa si possono muovere attraverso due tecniche di cinematica, inversa o diretta, usate singolarmente oppure mescolate. La cinematica diretta (FK) consiste nel muovere osso per osso fino a costruire la posa desiderata del personaggio, compiendo una serie di rotazioni per ogni giunto considerato. La cinematica inversa (IK) permette invece di creare delle pose utilizzando delle *maniglie* che muovono catene di giunti: quello che permette di interpolare la rotazione di più ossa insieme è chiamato *end effector* ovvero è il punto pivot dell'osso figlio selezionato alla fine di una catena cinematica. Ogni tipo di cinematica risponde ad esigenze particolari, ad esempio si può usare l'IK per creare l'animazione diretta di una mano che raggiunge un oggetto, mentre con la FK si può correggere la rotazione della spalla del personaggio mentre compie quell'azione.

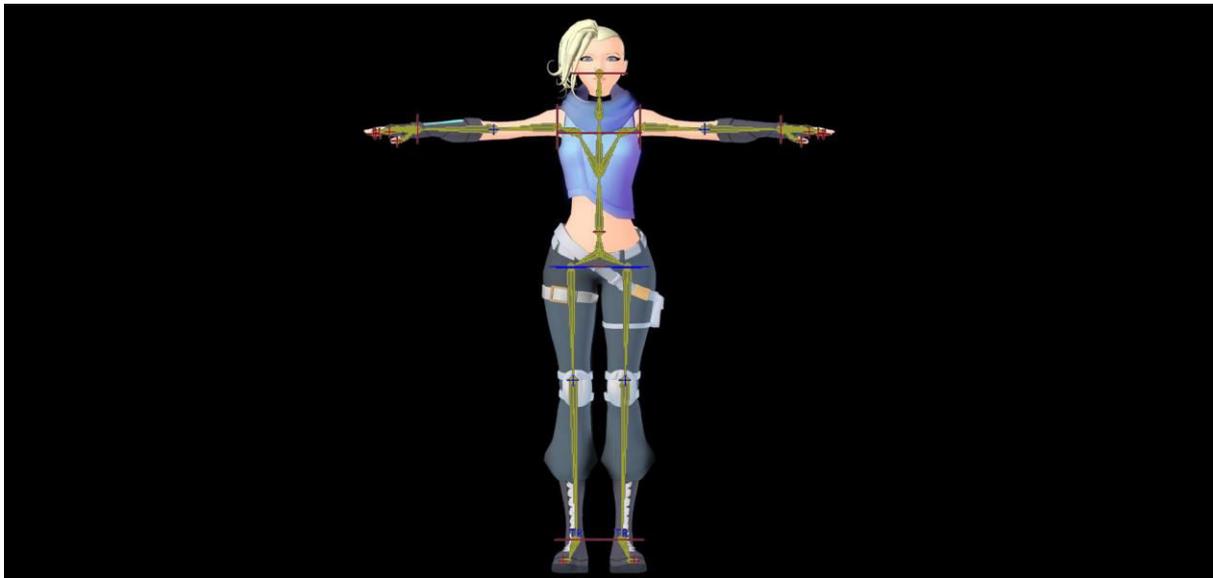


Figura 10.3: Esempio di scheletro per un personaggio 3D con visualizzazione IK (in rosso e blu) e FK (in giallo)

Riggare un volto aggiunge al gioco ulteriori gradi di difficoltà dovuti alla complessità della deformazione durante le espressioni facciali. E' compito del rigger costruire tutta la libreria di espressione che l'animatore sfrutterà tramite apposite interfacce, significa che la costruzione di tutte le possibili espressioni di un personaggio deve essere studiata e costruita in questa fase. Molto spesso per animare un volto non bastano semplici catene di ossa, e bisogna

intervenire applicando dei controllori su aree specifiche del volto. Si pensi all'azione di un sorriso: apparentemente l'azione che si svolge è quella di alzare gli angoli della bocca verso l'alto, ma questo movimento attiva tutti i muscoli adiacenti alla bocca, facendo alzare e gonfiare le guance fino a strizzare le palpebre inferiori, e creando delle fossette intorno alla bocca più o meno accentuate. Viene naturale quindi pensare di creare un controllore agli angoli della bocca che permetta di simulare il movimento appena descritto, semplicemente muovendo quel singolo punto. Nella Figura 10.4 vengono mostrati gli strumenti visuali che l'animatore ha a disposizione per mettere in posa il personaggio: a destra viene creata un'espressione muovendo i rombi (che sono legati a dei *drivers*) all'interno della loro area di azione. L'animatore lavora su questa interfaccia, mentre il rigger si occupa di programmare tutti i vincoli tra i vari comandi per ottenere queste deformazioni.

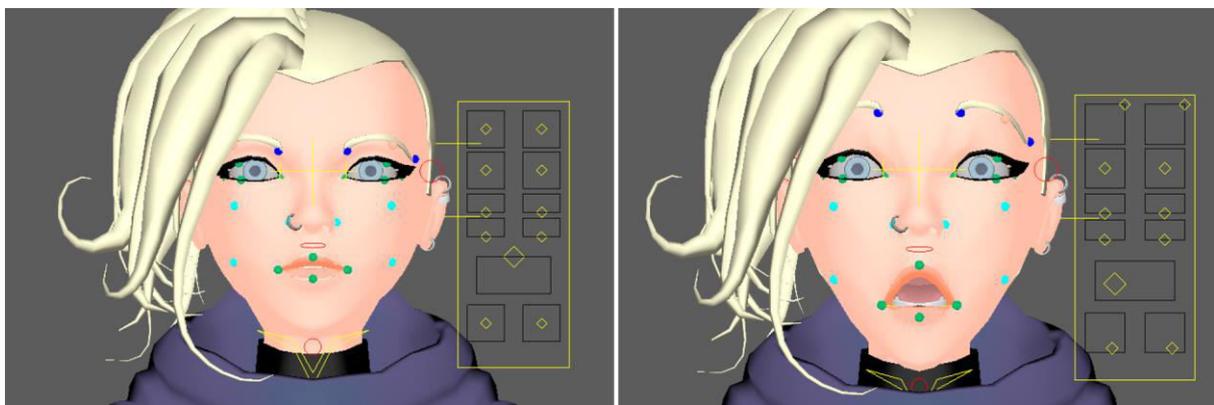


Figura 10.4: UI dei controllori per il rigg facciale del personaggio.

Si uniscono diverse tecniche nel rigging di un volto: alla creazione delle singole ossa, ad esempio per la mandibola o per le palpebre si unisce l'utilizzo dei lattices, che sono delle vere e proprie griglie di deformazione. Il *Lattice* è uno speciale oggetto 3D: visivamente, la forma è simile a quella di una mesh box, ma non è visibile nel rendering, lo scopo è quello di deformare una mesh basandosi sulla posizione dei punti di controllo. A questo si aggiunge la creazione di *blendshapes*, oppure *shape keys* o chiavi di forma, che permettono di richiamare in maniera più immediata delle pose specifiche rendendolo uno strumento indispensabile, ampiamente utilizzato nella creazione della libreria di espressioni.

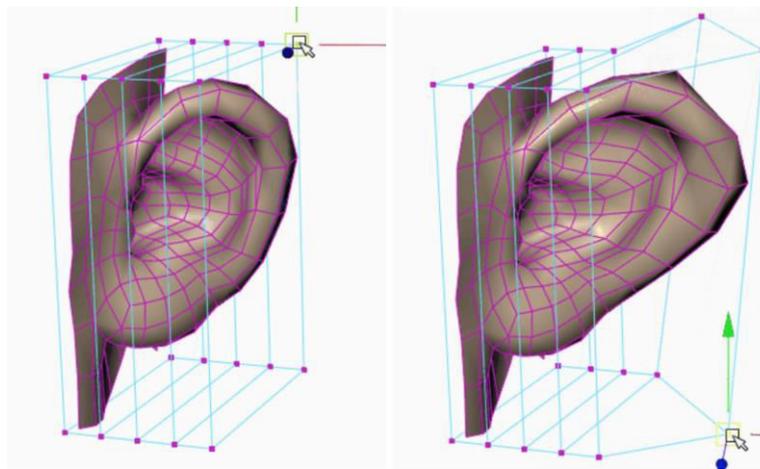


Figura 10.5: Esempio di deformazione guidata dal lattice

La fase di rigging per il personaggio di Nadya si è distinta in tre momenti separati: come primo si è scelto di lavorare sul corpo e successivamente di gestire il rigging del viso. Per ultimo si sono riggati i capelli per poter simulare l'animazione delle ciocche.

Ho applicato tre diverse tecniche, una per ognuno dei processi, utilizzando diversi tool con l'obiettivo di costruire un sistema complesso e animabile con un alto grado di libertà nei movimenti. A ogni più piccolo traguardo raggiunto nel metodo si è accostato un confronto con lo scheletro mocap e i dati prodotti dalla tuta inerziale al fine di garantire una completa compatibilità tra i due rig.

Quando è stato presentato il primo esperimento di retargeting dai dati skeletal della tuta, si era parlato di retargeting manuale che portava con sé qualche difficoltà nel vincolare correttamente le due armature. Da allora, proseguendo negli studi durante la produzione del trailer, si è potuto approfondire l'argomento del rigging, che ha permesso di acquisire delle capacità più eleganti e corrette nell'esecuzione del procedimento. Sono state effettuate diverse prove di rigging, tramite *Mixamo* e andando poi ad aggiustare i pesi dei vertici; con lo strumento *Quick Rig* di Maya per la definizione di scheletri veloci e non troppo complessi, e con lo script *Advanced Skeleton* delegato in seguito al rigging del solo viso. In previsione dei cambiamenti nella topologia del personaggio, è stato necessario trovare uno strumento in grado di trasportare lo scheletro creato per Nadya da una mesh all'altra, senza perdere la possibilità di effettuare il retargeting dai dati mocap: la risposta a questa necessità è stato *HumanIK*, un tool interno al software di Maya che permette di configurare correttamente uno scheletro e di facilitare l'interfaccia tra i diversi scheletri durante il retargeting, in maniera non distruttiva.

10.2.1 HumanIK per il rigging del corpo

All'interno del programma di rigging è possibile creare da zero uno scheletro per qualsiasi personaggio umanoide e di impostare sia la modalità FK e IK per il sistema. Inizialmente bisogna assicurarsi di avere il personaggio nella consueta T-pose rivolto verso l'asse -z positiva e l'asse verticale -y rivolta verso l'alto, è buona norma quindi muovere il pivot del personaggio sui piedi e allineare il pivot e la geometria all'origine degli assi. Per assicurarsi di non trovare spiacevoli sorprese durante la fase di animazione, è utile neutralizzare tutte le trasformazioni di traslazione, rotazione e scala del personaggio (valori [0,0,0] per traslazione e rotazione, [1,1,1] per lo scalamento): il comando *Freeze Transformation* () assicura il reset di tali valori alla posizione attuale.

Nella finestra di rigging è possibile aprire l'interfaccia *HumanIK* () e iniziare con la creazione dello scheletro, che viene inizialmente generato con delle impostazioni di default quindi il primo passo è scalare lo scheletro alla stessa dimensione del personaggio. Per il modello di Nadya, che di base sarà guidato dalle animazioni della Smartsuit, si sono impostati manualmente alcuni valori per avvicinare la configurazione dello scheletro a quella generata dai dati mocap. Sono stati dati per i giunti di *Spine* e *Neck* un valore pari a 2, ovvero entrambi gli elementi sono formati da una catena di due ossa ciascuno. Osservando lo scheletro mocap si può notare come altra differenza la presenza di un osso aggiuntivo in prossimità degli arti inferiori, quindi sul nostro modello si è aggiunto un osso di tipo *Toe Bones > Middle* con un numero di ossa pari a 1 e rimuovendo la spunta dalla voce *toe-based*: in questo modo le ossa degli arti inferiori combaciano con quelle del rig mocap. Il resto dei giunti è impostato correttamente quindi il passo successivo è quello di far cambiare le ossa alla fisionomia del corpo, il cui risultato è mostrato in Figura 10.6. E' utile avvalersi degli strumenti offerti da Maya, e in questo caso si sta parlando dell'interfaccia grafica che mostra la *definizione del personaggio* e il suo status: ogni volta che si muove un giunto, l'icona corrispondente si colorerà di giallo qualora l'osso non sia più parallelo agli assi. E' inoltre estremamente importante che i giunti presentino l'orientamento dei triassi in maniera coerente, onde evitare errori di animazione che costringerebbero a ripetere il rigging per andare a correggere le rotazioni errate. Fintanto che l'interfaccia mostra tutte le icone verdi, si ha la sicurezza di procedere correttamente (vedere esempio grafico della UI in Figura 8.6).

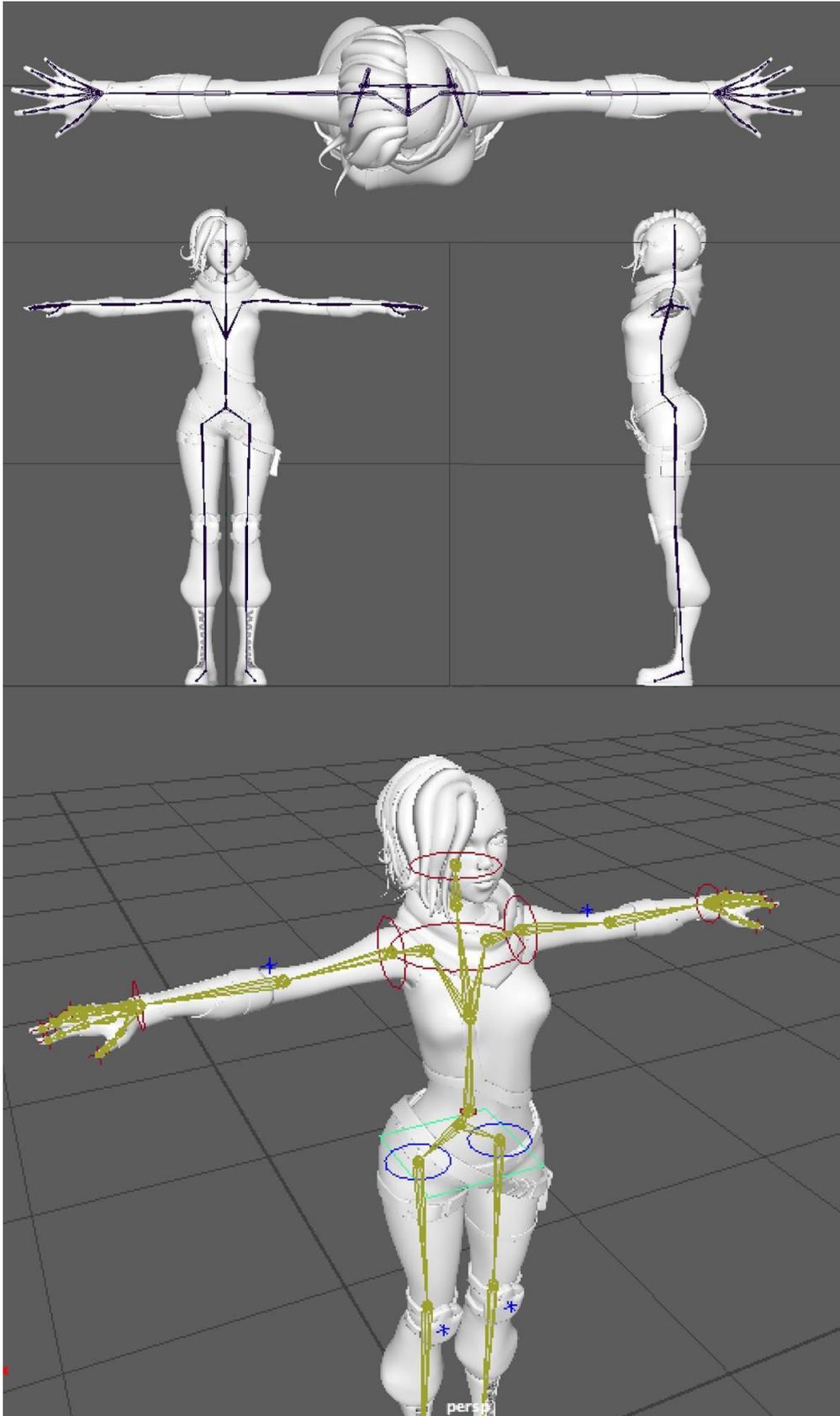


Figura 10.6: Impostazione dello scheletro visto dall'alto, frontalmente e di lato (sopra). Risultato del rigging, con la creazione di un sistema IK/FK per il personaggio di Nadya (sotto)

E' sufficiente lavorare su un lato dello scheletro perché esso combaci con la struttura del corpo, perché HumanIK dispone di un tasto (👤) che effettua il *mirror* del rig in modo che sia perfettamente simmetrico. Per legare il rig al modello bisogna effettuare il *binding*, disponibile nella voce menù **Skin > Bind Skin > ☐** che è accessibile impostando lo spazio di lavoro di Maya nella modalità rigging. Le opzioni che sono state scelte per il binding del modello di Nadya sono le seguenti:

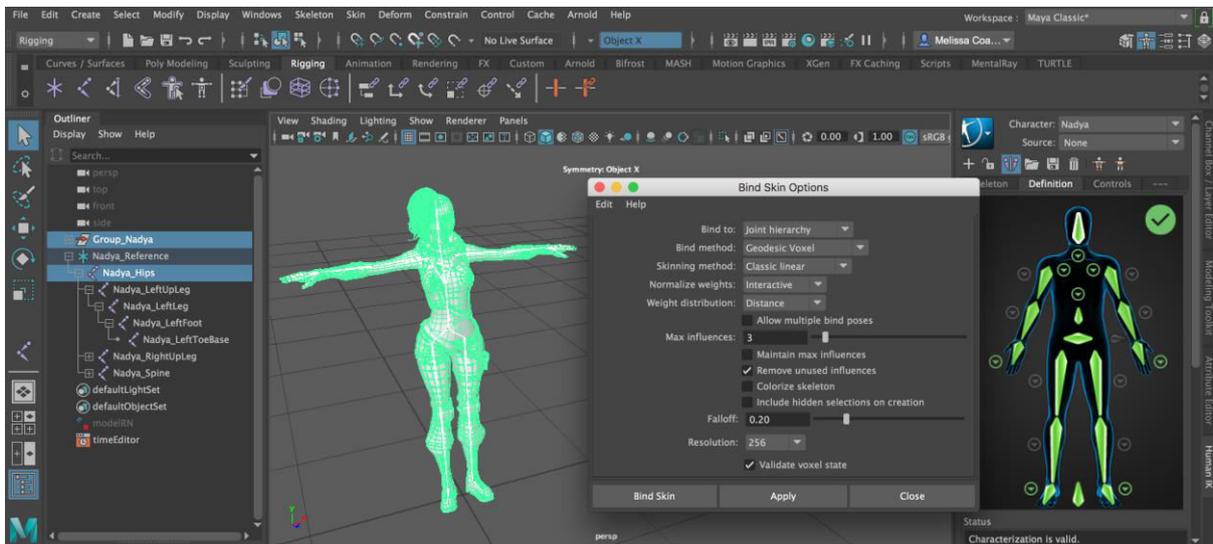


Figura 10.7: Finestra di selezione delle opzioni di vincolo del rig

- **Bind to: *Joint Hierarchy*** - Specifica che gli oggetti deformabili selezionati saranno legati all'intero scheletro, a partire dall'articolazione della radice, anche se è stata selezionata una giuntura diversa. Legare l'intera gerarchia è il metodo consueto che si usa per il binding della mesh di un personaggio.
- **Bind method: *Geodesic Voxel*** - Questo metodo di vincolo consente di selezionare più mesh, il che conferisce una ponderazione continua attraverso la geometria poiché tutti i poligoni appartenenti a geometrie differenti vengono considerate come un singolo volume. I calcoli del legame geodetico voxel influenzano i pesi che sono compatibili con il blend lineare, o il dual-quaternion skin scelto nel nostro caso.

Per essere utilizzato correttamente, l'associazione Geodesic Voxel ha due requisiti che devono essere rispettati:

- 1) a differenza di altri metodi di collegamento in Maya, con il binding Voxel geodetico è necessario racchiudere tutte le influenze dei giunti all'interno del

volume della mesh. Se ciò non viene fatto, la distanza geodetica non verrà calcolata correttamente;

- 2) le normali di tutte le facce della geometria devono essere orientate in modo che puntino verso l'esterno. Un modo semplice per controllare ciò è abilitare il comando *backface culling* nella viewport e ispezionare visivamente i risultati.
- **Skinning Method: *Dual Quaternion*** - Il metodo è progettato per eliminare effetti di deformazione indesiderati: l'opzione lineare è il metodo più semplice per creare una mesh che segue le giunture di un personaggio, ma in aree come il polso o il gomito ci si può imbattere in una perdita di volume della mesh mentre un osso si attorciglia sul proprio asse. Dove lo skin lineare porta la una mesh a collassare o restringersi quando un osso si attorciglia, il metodo a quaternioni aiuta a preservare il volume nella mesh, creando una deformazione più realistica in quelle aree problematiche.

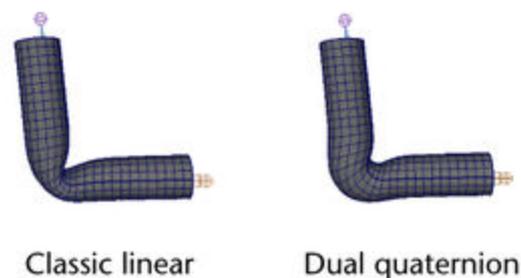


Figura 10.8: Deformazione della mesh secondo i metodi Classic linear e Dual quaternion¹

- **Normalize Weight: *Interactive*** - I valori dei pesi sono normalizzati in modo che quando si modifica un valore di peso gli altri vertici influenzati dalla stessa area di deformazione vengono automaticamente aggiornati. Quando la normalizzazione è attiva, Maya considera i pesi di tutti i punti della pelle interessati da una particolare area di influenza in relazione l'uno con l'altro, in questo modo si garantisce che tutti i valori dei punti della pelle continuino a sommarsi e ciò consente a Maya di tenere traccia della loro influenza relativa.
- **Weight Distribution: *Distance*** - I nuovi pesi sono calcolati in base alla distanza del vertice dalle varie influenze alle quali è vincolato, dove articolazioni più vicine hanno un peso maggiore.
- **Max Influence: 3** - Questo numero specifica quanti giunti possono influenzare una regione di vertici. Rispetto al valore di default (5) è stato diminuito di due passi per

¹ [URL_img](#)

avere maggiore controllo sulla deformazione di giunti in situazioni particolari come il gomito o le ginocchia.

Appena il calcolatore termina di effettuare lo skinning, il personaggio è ufficialmente animabile nel corpo, tramite controllori FK. Per incrementare la qualità, e la complessità, dello scheletro si può creare un *Control Rig* () che genera tutti i controllori IK insieme ad alcuni comandi per mescolare le due tecniche di animazione. Si ottiene un risultato come mostrato in Figura 10.3.

Tutte le opzioni descritte sono state scelte secondo una metodologia di testing diretto delle varie funzionalità offerte dal programma, e sono state successivamente confrontate in uno studio teorico e pratico con alcuni script di rigging più avanzati scoperti in rete, tra i più utilizzati dalle community (es. Advanced Skeleton e CDK Autorigger). Human IK è stato in grado di offrire la soluzione più semplice e più facilmente gestibile in termini computazionali e di complessità del sistema creato. Grazie al *Dual Quaternion* per esempio, il processo di weight painting per correggere manualmente le deformazioni è stato più veloce grazie a correzioni già effettuate dall'algoritmo usato dal metodo. Inoltre, l'interscambio di informazioni tra lo scheletro Humank IK e il rig generato da Rokoko si dimostra molto semplice: è sufficiente creare una *definizione del personaggio* per lo scheletro mocap e si può quindi scegliere di pilotare il rig del personaggio tramite i movimenti dell'altro scheletro invece che dal suo Control Rig. Quando si vuole tornare all'impostazione iniziale mantenendo le animazioni mocap, è possibile effettuare un *bake* dell'animazione sul Control Rig, e riassegnarlo al personaggio.

Nella maggioranza dei casi, anche il rigging preparato più accuratamente ha bisogno di essere ritoccato perché per quanto il programma possa essere preciso, non è esule da errori. Di fronte ad un personaggio vincolato a un'armatura, il primo approccio è quello di animarlo come un burattino mettendolo in pose diverse e testando tutti i giunti, alla ricerca di deformazioni indesiderate. I punti critici si trovano solitamente dove c'è il rischio di compenetrazione degli abiti con il corpo del personaggio, o nei punti del corpo dove piccole porzioni di vertici vengono pilotati da ossa adiacenti, come nel caso delle dita. Per correggere tali deformazioni si può intervenire manualmente con il pennello di *paint skin weight* (): si seleziona la mesh sulla quale si vuole intervenire e l'interfaccia di Maya mostra tutte le aree di influenza suddivise gerarchicamente come lo scheletro. Utilizzare una tavoletta grafica può

rendere il lavoro più certosino, grazie alla sensibilità del tocco della penna rispetto al clic del mouse. Un caso presentatosi durante questo lavoro che ha richiesto una correzione mirata sono state le dita, che piegandosi svelavano diversi errori di skinning: da un lato la non conservazione dei volumi delle falangi durante il piegamento del dito, e in alcuni casi addirittura alcuni vertici di un dito venivano associati al giunto di un altro dito adiacente.

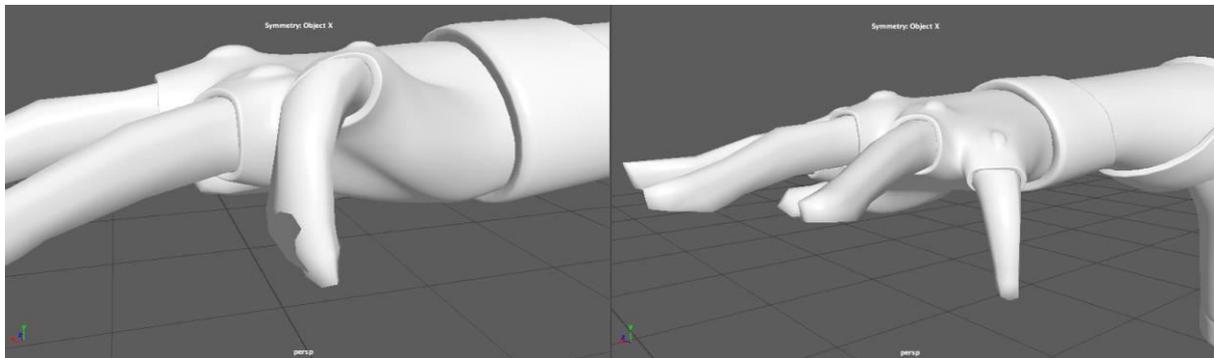


Figura 10.9 (a): Correzione dei vertex weights sul mignolo della mano sinistra.

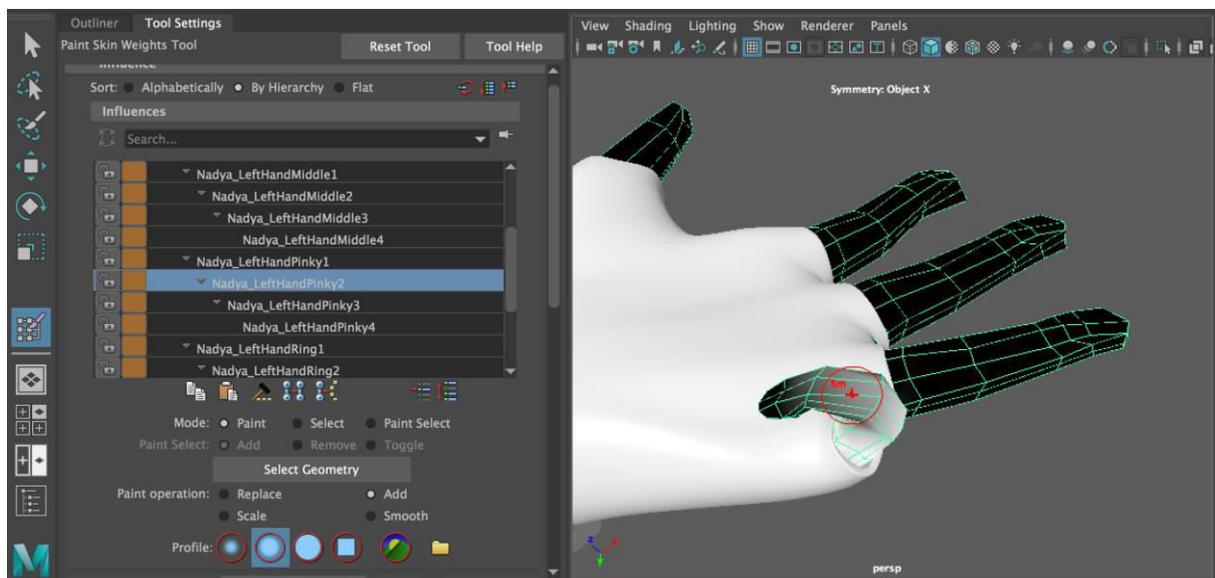


Figura 10.9 (b): Interfaccia del vertex weight painting, in bianco viene mostrata l'area influenzata dal giunto selezionato, con il pennello si può modificare la percentuale di influenza locale per ogni osso.

Per ottenere delle buone deformazioni è giusto che aree di vertici siano pilotati da più giunti con percentuali di influenza diversa, questo fa in modo di ottenere una deformazione elastica e più naturale, e l'osservazione della realtà rimane sempre la primaria fonte nello studio del realismo nel digitale. I principali interventi di *weighting* sono stati localizzati sugli indumenti di Nadya, poiché la parte superiore del corpo è vestita da tre strati di indumenti sopra lo strato della pelle: un top sportivo nero come base, un giubbotto sui toni del blu e una sciarpa. E' stata prestata particolare attenzione nelle aree localizzate attorno alle spalle e al collo in modo

da eliminare il più possibile errori di compenetrazione tra questi oggetti. Per il trailer non è stata richiesta una simulazione degli indumenti in quanto le simulazioni richiedono alte prestazioni di calcolo, e quindi di memoria, mentre tutto il progetto si è svolto nell'ottica del risparmio computazionale destinato a una maggiore velocità di rendering. Perciò è stato sufficiente prestare cura al peso dei vertici assegnati in quanto gli indumenti si comportano come degli oggetti semi-rigidi (inteso non come 'non-deformabili' ma come 'non-simulati').

Un caso di analisi differente è stato affrontato con quelle parti propriamente rigide degli accessori di Nadya, come le cinture: lo skinning ha prodotto degli effetti indesiderati perchè gli oggetti vengono trattati come deformabili nel loro intero, ma parti come le fibbie dorate delle cinture sono di materiali rigidi e non devono deformarsi insieme alla pelle. A questo proposito il problema è stato risolto utilizzando il comando *Wrap*, dopo aver svincolato l'accessorio dall'armatura. Con i deformati di wrapping, si possono modellare oggetti deformabili con curve NURBS o altri oggetti poligonali, che sono oggetti di influenza di avvolgimento. Questi vengono utilizzati come nodo dell'algoritmo per deformare un oggetto. La forma e le trasformazioni dell'involucro influenzano gli oggetti e i loro punti forniscono la forma della deformazione. Quando si crea un oggetto di influenza di avvolgimento Maya crea una copia dell'oggetto, lo si può ritrovare in elenco nell'outliner, e la usa come forma di base per la deformazione. Qualsiasi differenza di posizione, orientamento o forma tra la forma della base e l'oggetto di influenza dell'involucro determina una deformazione della superficie influenzata dal wrapping. Si possono combinare più oggetti di influenza per deformare in maniera concorrente l'oggetto rigido. Nella Figura 10.10 si dimostra come l'utilizzo del wrap deformer abbia migliorato in maniera significativa il risultato della deformazione delle cinture durante il piegamento delle gambe del personaggio.

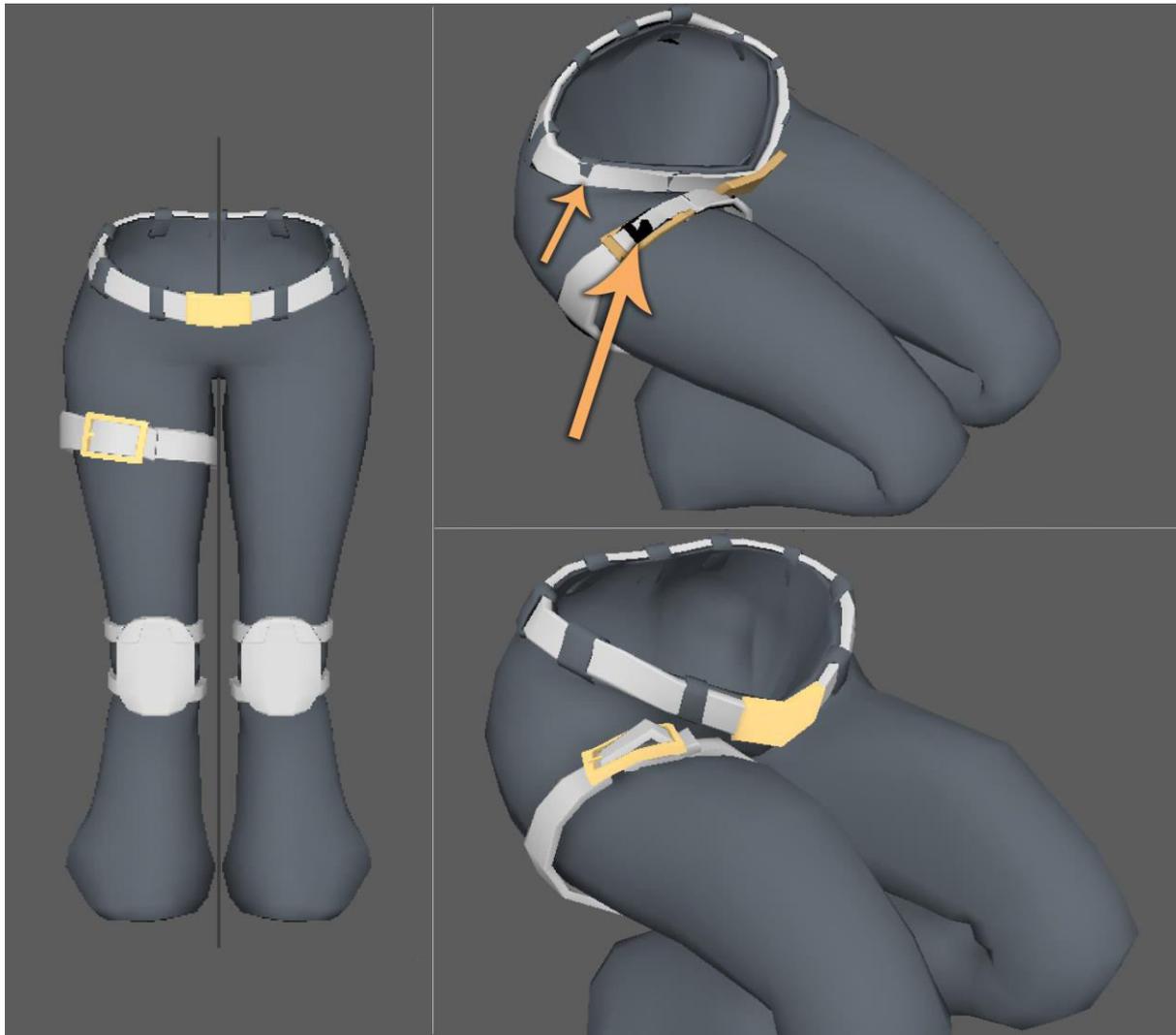


Figura 10.10: In alto le due cinture sono vincolate alle ossa del rig e piegando le gambe si ottengono degli effetti di deformazione errati per gli elementi come la fibbia ed i passanti della cintura. Nella parte di sotto si può apprezzare un netto miglioramento nella deformazione grazie all'utilizzo del deformatore Wrap: le cinture sono vincolate alla mesh in maniera rigida, con il risultato di una deformazione più corretta fisicamente..

Prima di passare al rigging del viso e dei capelli, si vogliono richiamare e condividere schematicamente gli step utilizzati durante il processo di rigging per il corpo di Nadya utilizzando il tool *Human IK* di Maya. Questo schema serve anche per fissare una scaletta standard da seguire per tutti i personaggi futuri che saranno animati dal sistema di motion capture adottato nella nostra pipeline, e getta le basi per la possibile creazione di uno script per automatizzare alcuni dei passaggi che non richiedono un intervento manuale, come il weight painting:

1) *Create Skeleton*

- a) Aggiustare la scala dello scheletro per combaciare con il corpo;

- b) Impostare i valori '*Spine = 2, Neck = 2*';
 - c) Impostare il nodo '*Toe Bones > Middle*' (n.bones = 1, toe-based unchecked);
 - d) Aggiustare le osse sul corpo in modo che l'interfaccia del character definition resti sempre tutto verde;
- 2) Binding tra scheletro e modello (nota al paragrafo successivo: se la testa è già riggata, selezionare tutto il corpo tranne la testa)
- a) Bind to: *Joint Hierarchy*;
 - b) Bind method: *Geodesic Voxel*;
 - c) Skinning Method: *Dual Quaternion*;
 - d) Normalize Weight: *Interactive*;
 - e) Weight Distribution: *Distance*;
 - f) Max Influence = 3;
- 3) Se la testa è già stata riggata, selezionare l'osso Head_M e unirlo al rig tramite il comando *Connect Joint*. Cancellare il giunto Head del rig intero. Assegnare il nuovo giunto Head alla definizione del personaggio.
- 4) *Create Control Rig* per genere l'IK;
- 5) Testare e sistemare i pesi con il comando *paint skin weight*;
- 6) Assegnare il modificatore *Wrap* agli oggetti rigidi che seguono la mesh ma non si deformano in maniera elastica.

Il tool di Human IK non dispone di una funzionalità per il rigging del viso quindi sono state vagliate diverse alternative per ottenere la migliore qualità per l'animazione facciale, tenendo conto degli strumenti di motion capture facciale che si sono utilizzati per la registrazione delle espressioni del viso e del monologo. Costruire un rig da zero avrebbe richiesto una mole di lavoro che in un tempo di produzione così ristretto non sarebbe stata efficiente. Si sono studiati algoritmi alternativi che potessero agevolare il lavoro per produrre un rig completo e la scelta è ricaduta su un tool già discusso e testato durante le fasi di questo lavoro di tesi, ovvero lo script Advanced Skeleton: è stato il naturale proseguimento per poter approfondire le nuove competenze acquisite, anche in funzione del retargeting facciale.

In definitiva si è ottenuto uno scheletro di giunti come mostrato in Figura 10.11, e la gerarchia è perfettamente compatibile con quella della tuta di motion capture adottata nella pipeline.

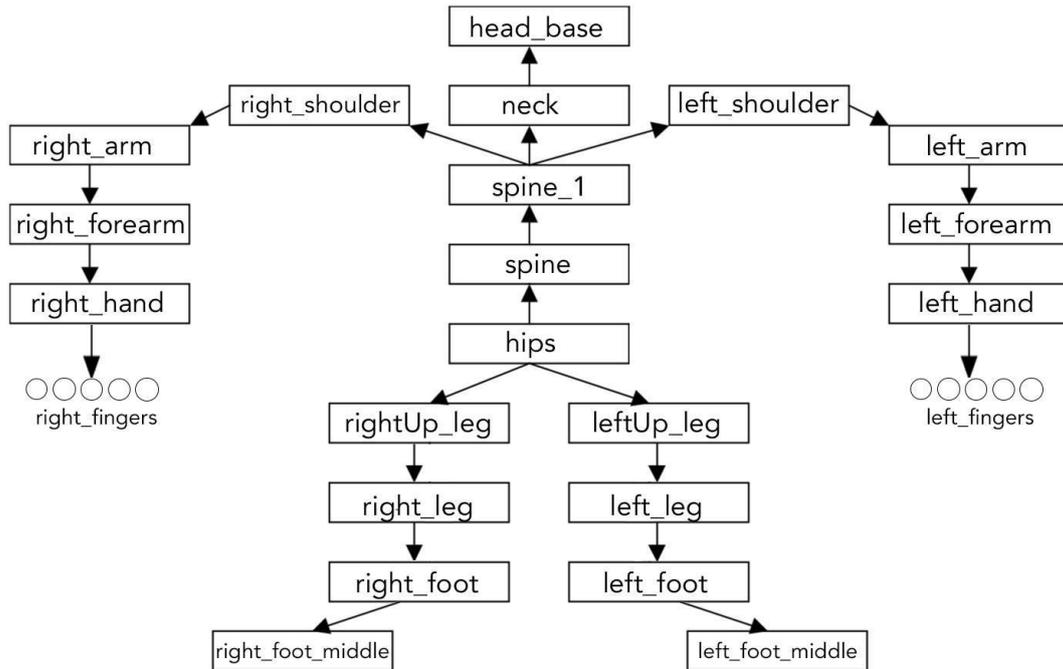


Figura 10.11: Struttura gerarchica del rig di Nadya

10.2.2 Advanced Skeleton per il rigging facciale

Advanced Skeleton è un tool creato nel 1996 da un'azienda denominata semplicemente Animation Studios, così si legge dal loro sito ufficiale, con lo scopo di offrire uno strumento agli animatori di creazione modulare di un rig. L'interfaccia del programma è suddivisa in schede dedicate al setup del personaggio, ognuna indirizzata a uno scopo preciso: seguendo passo passo si viene guidati alla creazione del rigging del corpo e successivamente del viso, all'impostazione di sistemi di driver e blendshapes. E' un tool completo che sfrutta le funzionalità di Maya con un'interfaccia dedicata e automatizzata nel processo di creazione, per venire incontro alle esigenze soprattutto degli animatori indipendenti con una scarsa conoscenza in materia di rigging. Ogni passo di lavoro è descritto nella sua funzione e documentato con video tutorial su Youtube, offrendo la possibilità di studiare il motivo alla base di alcuni passaggi e non solo un utilizzo automatico dei bottoni. Una peculiarità del programma che si è imparata durante il rigging facciale è che utilizza notevolmente la simmetria delle forme, specie nel viso dove effettua propriamente il mirror del lato destro su quello sinistro. Sono presenti alcuni bug a volte che alimentano la frustrazione del lavoro di animazione ma con alcuni accorgimenti e con un po' di training si è in grado di aggirare i diversi ostacoli e limitazioni nelle quali si può incorrere.

Nel nostro caso il rigging del corpo era già stato completato con lo strumento di Human IK, e siccome non ha richiesto sostanziali modifiche di progetto ho deciso di implementare Advanced Skeleton solamente nella creazione del rigging facciale, perché il monologo si sarebbe concentrato sul primo piano di Nadya ed è stato da subito individuata una priorità sul maggior livello di dettaglio del viso piuttosto che del corpo. All'inizio del progetto non era stato ufficialmente concordato l'utilizzo di motion capture per l'animazione del viso, e quindi requisito fondamentale per una buona animazione facciale era la possibilità di avere diversi tipi di controllori per gestire liberamente la creazione di pose di animazione. Una volta ottenuta la licenza di prova dei software Faceware Analyzer e Retargeter, è stato avviato il lavoro di mocap facciale per Nadya e gli stessi controllori si sono rivelati di grande aiuto durante la pipeline di lavoro, a conferma della stabilità e potenza dello script Advanced Skeleton.

Prima di iniziare il setup del viso è consigliato, e praticamente necessario, effettuare il cleaning del personaggio per assicurare una corretta topologia del modello e per cancellare dalla storia della mesh tutti quei nodi di attributi che possono creare conflitti durante la fase di rigging. L'interfaccia indica la pulizia del modello come primo step per l'avanzamento del lavoro: viene richiesta la creazione di un gruppo intitolato 'geo' che racchiude tutta la geometria del modello che si vuole riggare, dopodiché il gruppo viene esportato come file formato .obj. Tale formato è abbastanza contenuto nelle sue informazioni e con questo passaggio vengono rimosse informazioni come nodi inutilizzati, skin cluster e smoothing delle normali non applicati: di quest'ultimo abbiamo avuto un chiaro esempio in quanto nella successiva importazione della mesh pulita il modello appariva altamente squadrato, così come veniva visualizzato nei programmi di modellazione utilizzati dove venivano applicati degli modificatori per visualizzare la mesh liscia e uniforme. Grazie a questa scoperta casuale è stato possibile intervenire tempestivamente con il comando *Soften Edge* () , che ammorbidisce automaticamente gli angoli delle facce, prima della pulizia per garantire la conservazione della geometria.



Figura 10.12: Effettivo modello di Nadya dopo il cleaning del file. L'uniformità della geometria si recupera facilmente con il *Soften Edge* di Maya

L'importazione del file .obj pulito ha rimosso anche tutto il resto degli elementi del progetto che non appartenevano al gruppo 'geo', e per convenzione si è salvato il modello pulito in un nuovo file .mb (formato binary di Maya) e poi sono stati importati nuovamente nel progetto gli elementi precedenti, rig del corpo compreso.

Il primo input dello script crea un *fit skeleton* per il viso e l'utente deve trascinare i due cerchi rossi come mostrato nella figura tutorial, in modo che siano posizionati uno appena sopra la testa e l'altro sotto il mento. Viene richiesto di indicare le parti del viso che saranno utilizzare per il rigging, in particolare bisogna associare:

- *Face* - la mesh del viso;
- *All Head* - tutta la geometria che compone la testa, capelli e accessori compresi;
- *Left/Right Eye* - occhio destro e sinistro, comprensivi della membrana trasparente che racchiude l'occhio, se inclusa;
- *Upper/Lower Teeth* - dentatura e gengive superiori e inferiori;
- *Tongue* - lingua.

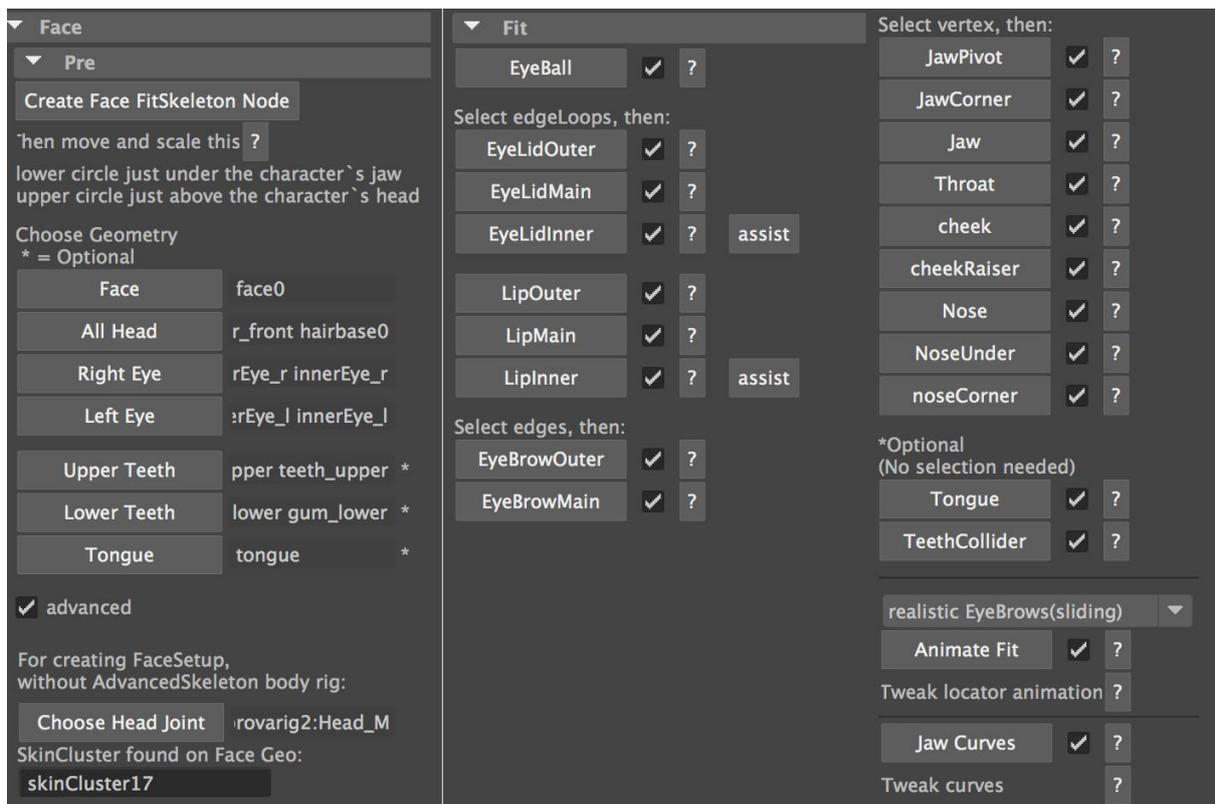


Figura 10.13: UI per la preparazione al rigging facciale, con tutte le voci da compilare

È importante evitare di modificare i nomi o i gruppi che compongono il viso in quanto si incorre in errori di generazione del rig. Durante queste impostazioni vengono creati degli skin cluster per ognuna delle regioni di interesse. Gli skin cluster sono un nodo per lo skinning che fungono da deformatore per legare la geometria alle articolazioni: in pratica sono gli elementi responsabili della suddivisione delle aree di influenza per i giunti durante la fase di skinning, e sono gli elementi che permettono di ridistribuire i pesi tramite il *weight painting*. Nelle opzioni avanzate si può notare che, come nel caso di utilizzo, è possibile riggare il viso anche in assenza di un rig del corpo generato da Advanced Skeleton. Per assicurare una perfetta integrazione del sistema con il rig Human IK è bastato sostituire l'osso *Head* con il nuovo *Head_M*, notificare la definizione del personaggio della modifica, e ricreare il control rig.

La sezione successiva riguarda l'identificazione delle regioni di deformazione per le ossa e per i driver, tramite la creazione di loop o di segnalazione di vertici, in questo modo si insegna al programma la topologia del viso in esame:

- *EyeBall* - viene generato un oggetto sferico che deve essere sistemato nella posizione dell'occhio;

- *EyeLid* (edge loops) - sono richiesti tre edge loop che identifichino rispettivamente il contorno delle palpebre, il contorno esterno dell'occhio e quello più interno;
- *Lip* (edge loops) - altri tre edge loops da posizionare rispettivamente sul conto delle labbra, sul centro, e all'interno della bocca nel punto dove terminano di fatto le labbra;
- *EyeBrow outer* (edge) - si delinea il profilo della fronte, è consigliato selezionare fino al lato che interseca la linea di vertici che porta al contorno dell'occhio, in modo da includere l'edge definito subito dopo;
- *EyeBrow Main* (edge) - definisce il perimetro del sopracciglio;
- *Jaw* (vertex) - viene richiesto di selezionare un singolo vertice in prossimità dell'inizio della mandibola, uno sull'angolo e uno sulla punta del mento, per tracciare il profilo intero dell'osso della mandibola;
- *Throat* (vertex) - si definisce il vertice in prossimità della gola;
- *Cheek* (vertex) - devono essere individuati il vertice degli zigomi, maggiore e minore;
- *Nose* (vertex) - si segnalano i vertici della punta del naso, della sua attaccatura, e del contorno laterale.
- *Tongue* - vengono generati i *locators* da inquadrare sulla lingua, in prossimità delle aree di deformazione;
- *Teeth Collider* - si crea una specie di schermo che andrà a limitare le collisioni tra bocca e pareti dentarie;
- *Animate Fit* - viene generata un'animazione del sopracciglio assegnato che verrà usato come driver per la simulazione N-cloth. Si può andare a correggere manualmente la posizione dei locator nei keyframe assegnati per fare in modo che non compenetrino nella mesh del personaggio;
- *Jaw Curves* - vengono create quattro curve che assegnano il profilo di deformazione della mandibola all'interno dell'area delimitata dalle curve più esterne, per pilotare tutte le animazioni facciali generate dal movimento del giunto *Jaw*.

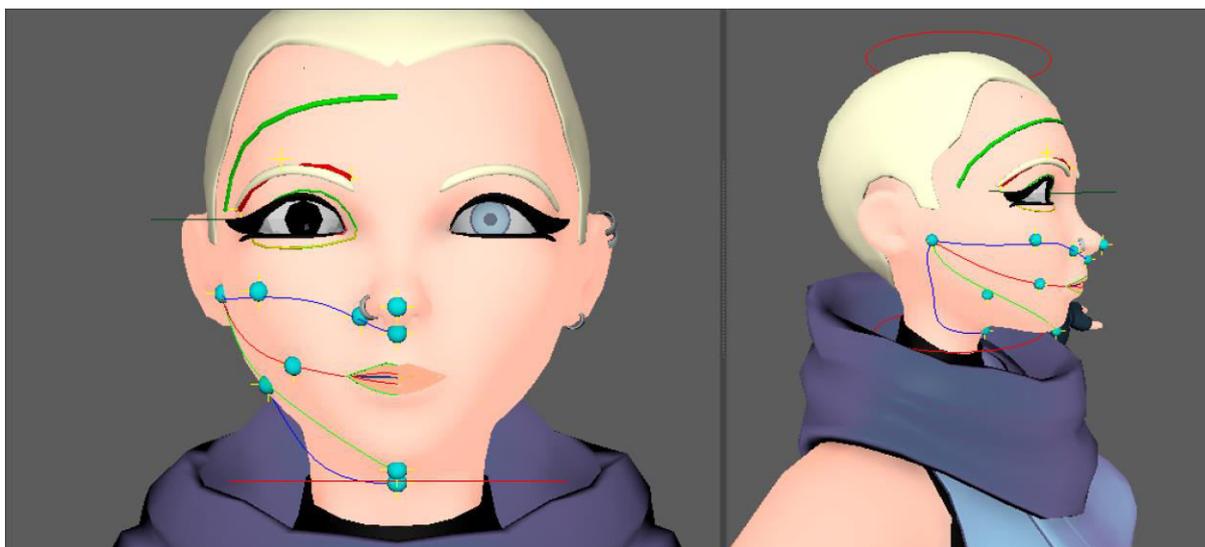


Figura 10.14: Risultato dell'impostazione dei parametri che lo script analizzerà per la creazione del rig facciale
(cliccare sull'immagine per andare al video)

E' di vitale importanza salvare una copia del file prima di procedere con la creazione dello scheletro del viso in quanto all'incorrere di un errore grave sarebbe necessario ricominciare il procedimento da capo. Lo script a questo punto offre due alternative per la creazione dello scheletro, una automatica e una composta da trenta passi che permettono di verificare man mano che lo si stiano utilizzando i dati giusti della mesh. Si sono testate entrambe le soluzioni e la seconda è risultata più affidabile in quanto si può intervenire localmente qualora si trovino vertici assegnati erroneamente (ad esempio all'area delle labbra erano stati assegnati in alcuni casi dei vertici situati al fondo del cavo orale). Sono stati necessari alcune prove di setup dei marker sul viso prima di trovare l'impostazione corretta per il modello di Nadya, ma il risultato è stato più che soddisfacente e si è potuto apprezzare un viso finalmente portato in vita. Per unire il rigging della testa allo scheletro precedentemente creato basta seguire i passi descritti in precedenza, ovvero usare il comando *connect joints* per connettere il giunto Head_M al corpo Human IK, e successivamente cancellare il giunto Head esistente, facendo attenzione di notificare la caratterizzazione del personaggio della modifica dell'osso. Inoltre è meglio non rinominare l'osso di Advanced Skeleton in quanto a fronte di modifiche necessarie, il programma cercherà di risalire al primo nome con cui l'ha individuato.

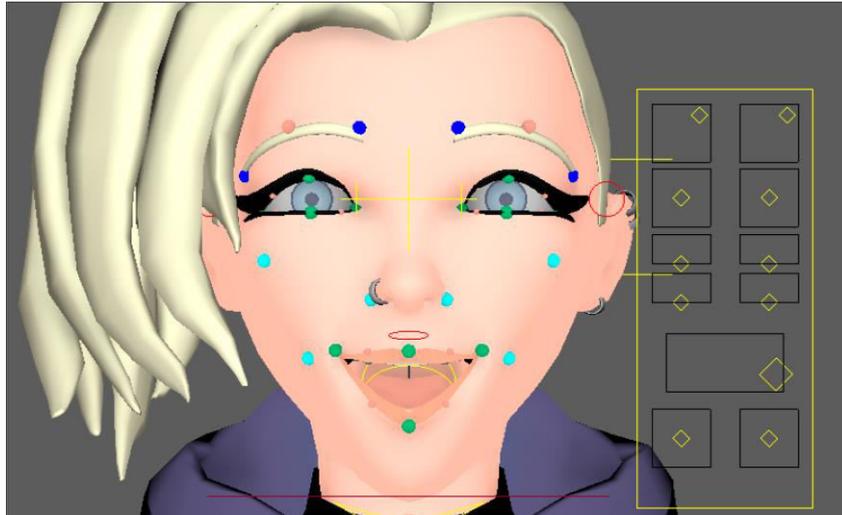


Figura 10.15: Interfaccia dei controller creati da Advanced Skeleton per l'animazione facciale.

Oltre ad aver creato tutta la struttura di ossa, il programma ha provveduto a creare tutti i controllori associati ai drivers delle blendshapes. Il comando *BlendShape* è l'implementazione di Maya della tecnica che permette a una singola mesh di deformarsi per ottenere numerose forme predefinite e qualsiasi numero di combinazioni di queste forme; in Blender sono altresì conosciute come 'chiavi di forma'. E' lo strumento principale di creazione della libreria di espressioni. Il viso impostato in maniera neutrale serve come base di partenza per la creazione di tutte le deformazioni, e le forme che dalla base vengono create si mescolano e si trasformano in espressioni più o meno complesse (risata, cipiglio, labbra increspate, naso arricciato, palpebre chiuse, sopracciglia alzate, forme della bocca a formare vocali, consonanti, ecc.). Questi sono i target dei blendshapes.

La collezione di forme espressive del viso è collegata alla forma di base neutra da un nodo di fusione, un operatore che pilota il modificatore. Al momento opportuno in un'animazione, l'animatore può quindi alzare la forma del volto appropriata necessaria e la forma facciale di base si animerà in una traiettoria lineare per assumere parzialmente o completamente la forma di destinazione, e formare l'espressione che è stata precedentemente modellata e collegata. Usando questo metodo, gli animatori possono mescolare e abbinare blendshape targets per formare un numero qualsiasi di combinazioni tra le forme di fusione preparate e collegate: un cipiglio può essere mescolato con un sorriso per formare un'espressione di scusa, e ancora, una blendshape con le labbra increspate può essere mescolata con gli occhi accigliati per formare un'espressione di determinazione o uno sguardo di disapprovazione. Maya tiene traccia di quanto ogni vertice è spostato rispetto alla

forma base originale, così quando l'animatore mescola o trasforma l'espressione, il software traduce quei vertici verso le posizioni finali come descritto da quel particolare blendshape.

Non è detto che Advanced Skeleton sia in grado di generare delle blendshape corrette con la prima configurazione, ma ci sono alcuni metodi che permettono di rielaborare le forme ed avere il pieno controllo sulle deformazioni. *Start Blendshapes wizard* passa in rassegna trentadue pose facciali, e l'utente può modificare aree specifiche per correggere compenetrazioni e deformazioni indesiderate: è sufficiente effettuare le modifiche per un lato del viso e lo script provvederà a riflettere le modifiche sull'altro lato.

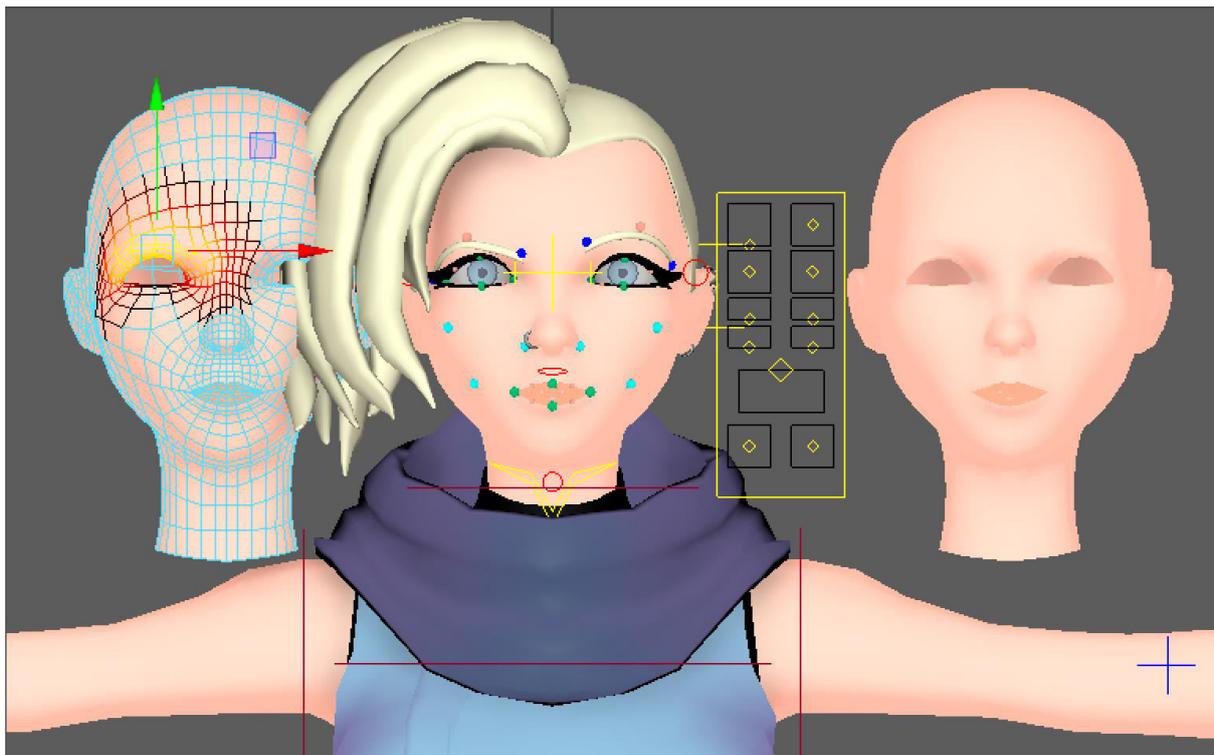


Figura 10.16: Interfaccia del *Blendshape Wizard* per la correzione delle espressioni facciali.

Qualora necessario è possibile crearne di nuove e assegnare un driver per pilotare la deformazione, ma nel caso di Nadya l'impostazione di base è stata più che sufficiente dal momento che sarebbe stata principalmente animata in motion capture. Il rigging facciale si rivela essere un punto delicato nella pipeline del progetto, basta un vertice appena fuori posto per mettere piede nell'Uncanny Valley, seppur vero che nei prodotti d'animazione è possibile concedersi diverse libertà, dipendentemente dall'esagerazione o dalla punta di realismo che si vuole imporre sul personaggio.



Figura 10.17: Pose library per alcune delle espressioni facciali di Nadya.

10.2.3 Dynamic Fk per il rigging dei capelli

I capelli sono una questione spinosa che ancora nel momento della finalizzazione della prima versione del trailer suscitano diverse incertezze. La problematica maggiore è il modello, non è ancora stata trovata una modalità che abbia convinto tutto il team di sviluppo. In quanto responsabile del progetto è stato mio dovere ascoltare le voci dei membri del gruppo e trovare la soluzione più congeniale per il tempo che avevamo a disposizione. Come si può notare ci sono stati diversi cambiamenti in corso d'opera rispetto alle immagini di Nadya consultabili all'interno di questa ricerca, il modello dei capelli è stato più volte rivisitato e attualmente somiglia a una anteprima come si vede in Figura 10.18.



Figura 10.18: Test modello nuova versione dei capelli di Nadya. (cliccare sull'immagine per andare al video)

Una volta approvato il modello da utilizzare per il trailer, si è potuto procedere con lo studio del rigging dei capelli, utilizzando catene dinamiche. Essendo una mesh unica, si è cercato il modo di gestire aree della geometria con catene di giunti differenti, principalmente uno per la parte frontale e uno per quella posteriore. La ciocca frontale è quella che sarà più in vista durante il teaser trailer quindi si è posta maggiore attenzione su quel dettaglio, mentre nelle ciocche di dietro ci si è limitati ad assegnare un leggero movimento guidato dai valori degli attributi *stiffness* e *mass* per simulare una dinamica più contenuta.

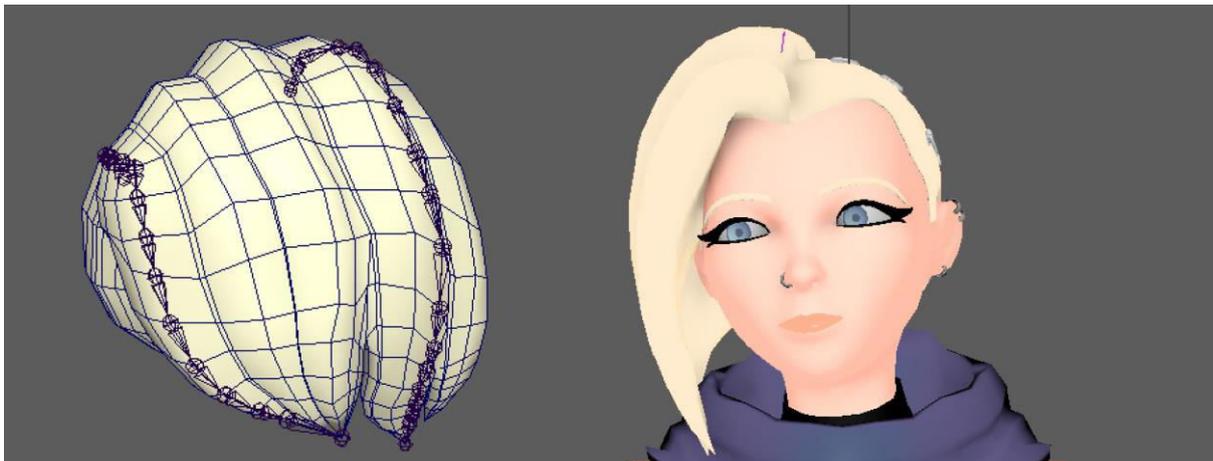


Figura 10.19: Posizionamento delle catene di ossa per la simulazione dinamica (cliccare sull'immagine per andare al video)

Sono state create due catene di ossa sulle quali viene costruita una *IK spline handle* (👁️), una curva che serve come base per creare dinamicità alle ossa. Nella finestra di Maya denominata FX, nella voce di menù *nHair* si trova appunto il comando per rendere la curva dinamica (🔗) e questo comporta la creazione di un sistema dinamico di capelli.

Con *nHair* si può simulare il movimento naturale di capelli lunghi, capelli che soffiano nel vento o vengono asciugati con il phon, varie acconciature tra cui coda di cavallo, trecce ecc., collisioni tra capelli e abbigliamento caratteriale, e altri effetti di curva dinamica come corde, catene, cavi, fili, un ponte sospeso, e addirittura creature marine. Entrando nello specifico, un sistema di capelli è una collezione di follicoli: una superficie NURBS o poligonale può avere più sistemi di capelli sulla stessa superficie e un sistema di capelli può estendersi su più superfici. Nella realtà comune, un follicolo dei capelli umani ospita tipicamente un capello, mentre in Maya ogni follicolo ospita una curva di capelli, ne controlla gli attributi e le curve associati a un particolare ciuffo di capelli, e imposta come i capelli si attaccano a una superficie NURBS o poligonale.

Nel caso dei capelli di Nadya, i capelli sono formati da un blocco unico che è attaccato a una radice rappresentata dall'oggetto *scalp*, e il nodo che gestisce tutte le informazioni di comportamento dinamico è il *nucleus*. All'interno del nucleo si inizia a definire il punto di blocco (point lock), che specifica dove i capelli sono attaccati rispetto alla posizione iniziale. Si è vincolata la base (come un capello normale), mentre altre opzioni sono entrambe le estremità (come una corda attaccata alle estremità), oppure senza attaccature (i capelli cadranno come una corda allentata che viene fatta cadere). Nel caso di compenetrazioni troppo evidenti, oltre ad attivare l'opzione collisione per il sistema di capelli è necessario stabilire un *passive collider* che coincide con l'oggetto con cui i capelli si scontrano.

Ci sono volute davvero molte prove tra modello e simulazione prima di ottenere un risultato soddisfacente che fosse naturale ma non troppo esagerato, per evitare il rischio di dover gestire diversi tipi di collisioni e per non far sembrare i capelli di Nadya come un'acconciatura super gonfia appena uscita dal parrucchiere.

10.2.4 Rigging di Alep l'insetto

Alep comparirà per qualche secondo all'interno del trailer, ed è stato quindi necessario crearlo per poterlo integrare nella scena. Il modello non ha rappresentato grossi problemi ed è stato realizzato in un lasso molto breve di tempo, per un totale di 5600 vertici. In questo per il rigging si è fatto buon uso dello script Advanced Skeleton perché mette a disposizione diversi set di rig adattabili per una serie di personaggi, in particolare è presente uno scheletro *bug* (insetto), perfetto per il nostro insetto. Generato il rig è necessario adattarlo alla T-pose del personaggio, aggiungendo alcune ossa in più nelle zampe posteriori per poter seguire la geometria di Alep: il modello in T-pose rappresenta come l'insetto tenderà a muoversi in scena, non userà tutte le zampe per camminare ma assumerà una posizione semi-eretta e il primo paio di zampe saranno un po' le sue braccia.

È stata creata manualmente una catena di ossa per il corno frontale, avendo cura di posizionarla perfettamente al centro della linea di simmetria: una volta in posizione si sono bloccati i valori di traslazione e rotazione sugli assi di interesse per fare in modo che non si spostasse dalla posizione 0, altrimenti il programma di generazione dello scheletro avrebbe trattato l'oggetto come uno da duplicare a specchio sull'asse-x.

10.3 Animazione

L'animazione per il progetto *'Reverie Dawnfall'* prevede l'utilizzo prevalente di motion capture come base per impostare uno standard qualitativo che con la sola animazione tradizionale richiederebbe un lasso di tempo esponenzialmente enorme, in considerazione della qualità che è in grado di definire il mocap. Il lavoro con motion capture stravolge la pipeline dell'animazione perché coinvolge stadi differenti e si inserisce nell'ottica dell'approccio top-down: i dati offerti dal motion capture producono un'animazione completa che è tanto precisa quanto la qualità del sistema in uso. In una situazione ideale l'animazione con il motion capture si può considerare un prodotto pronto per il rendering ma la realtà dei fatti dimostra come ci sia sempre occasione di pulire e rifinire i dati.

Il teaser trailer ha una durata di circa 90 secondi, a 24 fotogrammi al secondo significa che è composta da circa 2700 frame. Nell'ottica di lavoro top-down, il motion capture consente di ottenere un'animazione completa invece di creare manualmente i keyframe per ogni frame, sta all'animatore far rispettare la soglia di qualità richiesta dal prodotto ma è notevolmente agevolato. La traccia mocap può essere quindi considerata come un'animazione grezza da usare come base di partenza, o come reference per il blocking.

Durante il lavoro di produzione del trailer si sono riscontrati alcuni problemi tecnici che non hanno permesso di svolgere delle registrazioni in condizioni ottimali, il che ha aumentato notevolmente la quantità di polishing da effettuare sui dati in post-produzione. Durante le registrazioni non si è mai avuta una velocità di connessione stabile e costante e ciò ha influito pesantemente sulla qualità dei dati, rischiando di compromettere l'effettivo valore aggiunto dell'utilizzo di motion capture nella pipeline. Per quanto riguarda il mocap facciale, nonostante le precauzioni assunte per le riprese al massimo della stabilità, non si è potuto evitare un minimo di jittering dovuto a movimenti bruschi e ad una configurazione del caschetto normalmente più larga delle proporzioni di Maddalena.

Al risultato ottenuto si può affermare che il motion capture si è rivelato comunque essere un insostituibile alleato per ottenere delle animazioni di qualità soprattutto in tempi ristretti, dove la sola animazione keyframe-based non avrebbe potuto ottenere lo stesso livello di dettaglio come ad esempio nelle micro-espressioni o nei movimenti più naturali rappresentati con il corpo.

10.3.1 Performance Capture

Sono stati effettuate diverse sessioni di ripresa, per un totale di tre incontri: il primo si sono registrati solamente i movimenti del corpo mentre nei due successivi la registrazione è avvenuta contemporaneamente per corpo e viso, quindi full body, grazie all'utilizzo della tuta inerziale Smartsuit e di una GoPro assemblata su un caschetto debitamente modificato perché seguisse con fedeltà estrema i movimenti della testa. La qualità della performance con la tuta inerziale è stata differente in ognuna delle tre sessioni, come testimoniano i graph editor per il giunto *hips* esaminato nelle tre sessioni di registrazione:

- 1) La prima performance si è svolta in tranquillità, il framerate della tuta si è assestato intorno agli 80fps, un po' al di sotto della media ma costanti, e la registrazione esportata ha mantenuto una certa fluidità lungo tutta la durata dell'animazione. Come ritmo rimane più lenta rispetto alle sessioni successive infatti l'animazione sarebbe più lunga di 10 secondi, e anche i movimenti di Maddalena erano molto meno amplificati.
- 2) Durante la seconda registrazione si è verificata più volte la necessità di stoppare le riprese perchè la connessione era talmente altalenante da non riuscire a mantenere la connessione della tuta, con la conseguenza che l'azione veniva registrata a scatti o non registrata affatto. Un paio di take sono stati registrati al massimo del framerate (110fps) e senza bruschi cali, evitando così di cancellare una giornata di lavori.
- 3) La terza registrazione è stato un altro caso anomalo in quanto durante la registrazione il framerate segnalato era molto alto, mentre sullo schermo si ricevevano immagini a scatti che si sono ripercosse in un'esportazione di qualità non ottima.



Grafico 10.1: Confronto della bontà del grafico nel giunto *hips*, componente *translate-x*



Grafico 10.2: Confronto della bontà del grafico nel giunto *hips*, componente *translate-y*

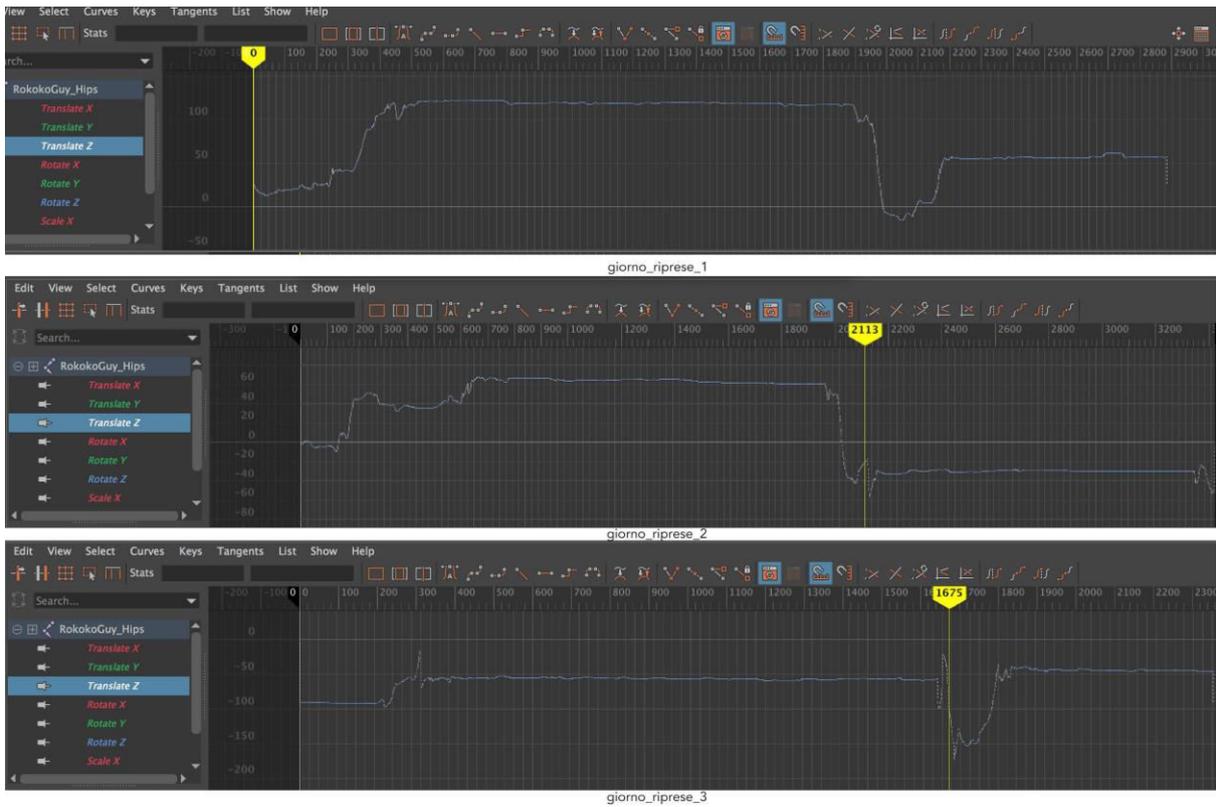


Grafico 10.3: Confronto della bontà del grafico nel giunto *hips*, componente *translate-z*



Grafico 10.4: Confronto della bontà del grafico nel giunto *hips*, componente *rotate-x*



Grafico 10.5: Confronto della bontà del grafico nel giunto *hips*, componente *rotate-y*

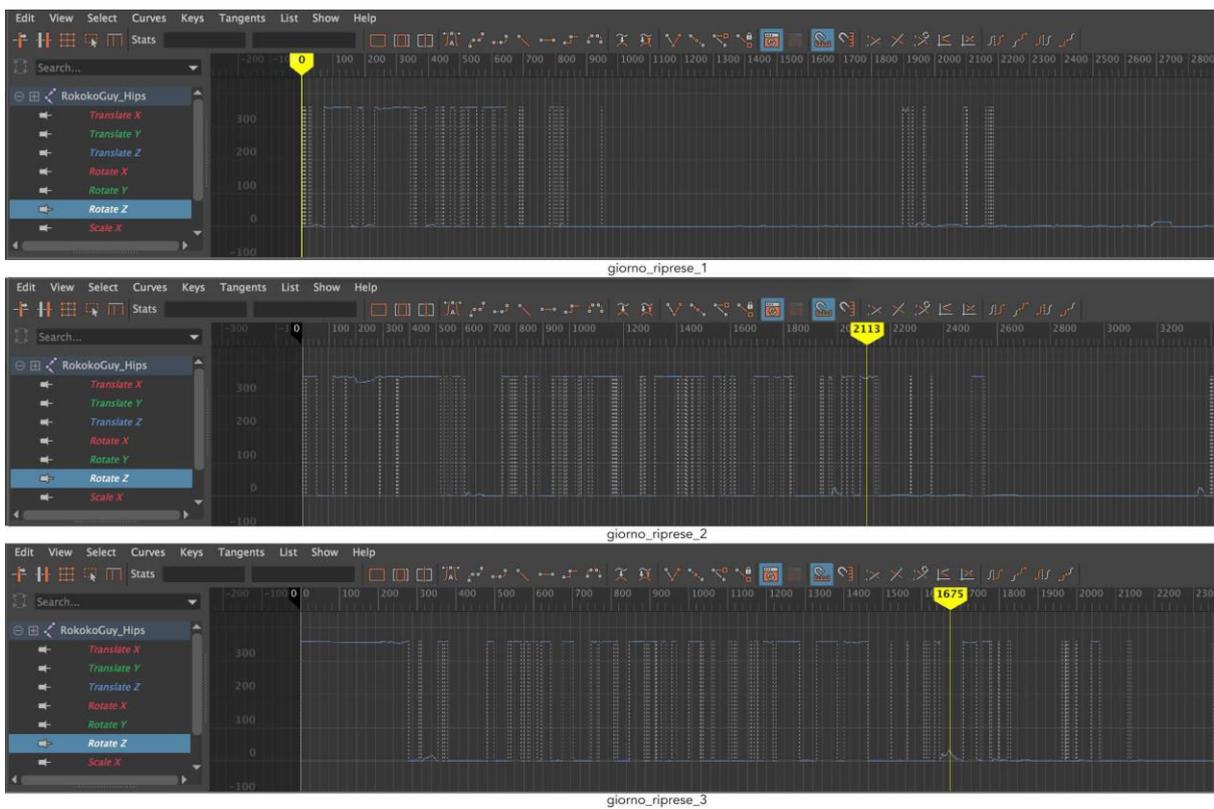


Grafico 10.6: Confronto della bontà del grafico nel giunto *hips*, componente *rotate-z*



Figura 10.21: Momenti di ripresa in motion capture durante le varie sessioni

Il motion capture facciale registrato durante gli ultimi due incontri non ha riscontrato problematiche se non quella di apportare qualche modifica al casco dopo il primo utilizzo per renderlo più stabile e conforme alla testa dell'attrice.

Seguendo un'attenta analisi sulle migliori esportazioni per ogni giornata di registrazione, ne è stata selezionata una per giornata, e si sono messe a confronto per studiare quale fosse la migliore ripresa in termini di pulizia dei dati e di performance di Maddalena. Dando uno sguardo ai grafici si può iniziare a valutare la qualità dei dati in relazione tra loro, ed uno dei primi aspetti che emerge è che la calibrazione durante il terzo giorno è nettamente differente rispetto agli altri due casi, come attestano i grafici della rotazione-y e traslazione-z. Come dimostrato nel caso studio di Rokoko, è sufficiente applicare un filtro di Eulero per affettare i grafici di rotazione dove la forma ad onda quadra è prevalente, in questo modo si è in grado di uniformare e correggere le rotazioni (qualora il filtro funzioni efficacemente).

La scelta della registrazione non è caduta su una singola ripresa in quanto in ognuna delle prescelte si trovano punti di forza che si è cercato di estrapolare e unire in un'unica animazione:

- 1) Dalla prima si apprezza il fatto di avere una stabilità globale della scena, soprattutto nei momenti in cui l'attrice si siede e si rialza. Da questa clip si è estratto il movimento di base che verrà successivamente ri-temporizzato.
- 2) Dalla seconda clip si sono estratti i movimenti degli arti superiori, poiché molto più espressivi ed esagerati dei precedenti, nel tentativo in cui la tuta ha funzionato quasi del tutto correttamente.
- 3) Dall'ultima giornata di riprese si è ricavato il motion capture facciale ufficiale perché il caschetto rifletteva un'ottima impostazione, con il viso perfettamente in asse e sempre in luce. Dalla registrazione del corpo si sono estratti altri movimenti che andassero a correggere inizialmente gli errori della clip precedente, come ad esempio il fatto che le mani non poggiavano terra quando dovuto, e per sincronizzare il tempo globale con la performance facciale.

L'unione di queste animazioni ha permesso di sfruttare tutte le giornate delle riprese e di testare l'interfacciarsi dei diversi file dati del motion capture prodotti da ognuna delle performance, ed ha permesso di studiare come integrare pezzi di animazione di motion capture diversi in un unico elemento di animazione basilare. Inoltre l'utilizzo di motion capture inerziale con le nuove tecnologie offerte permette davvero una libertà d'azione che

non è più vincolata a un volume di cattura, e in pochissimo tempo è possibile impostare delle riprese, letteralmente basta cambiarsi d'abito.

10.3.2 Retargeting del corpo

Per permettere al modellatore di continuare a modificare la topologia di Nadya, la levigatura delle animazioni mocap è avvenuto direttamente sullo scheletro *Rokoko_guy*, per poter lavorare in parallelo insieme agli altri reparti e senza rischiare di ripetere il lavoro da capo una volta avuto il modello definitivo. Non è l'unica strada percorribile, ma è quella che si è scelta perché sembrata la più coerente con la pipeline che si stava aggiornando: è anche possibile trasferire set di keyframe da un rig all'altro, ma mancando della praticità necessaria all'interno dell'ambiente di Maya, programma 3D che è stato scelto per lo sviluppo del trailer dopo aver passato gli ultimi anni accademici su Blender, si è preferito un approccio WYSIWYG (What You See Is What You Get, tradotto 'quello che vedi è quello che ottieni'). Inoltre è sempre stato possibile ad ogni aggiornamento testare il nuovo personaggio, perché grazie al tool di Humank IK il retargeting del corpo si effettua velocemente in pochi passaggi. Per questa ragione si è anche scelto di animare lo scheletro mocap, perché rispetto a trasferire i keyframe da un rig all'altro, effettuare il retargeting è sì la prassi per trasferire le animazioni mocap ma è anche un mezzo per iniziare a lavorare sulle animazioni in maniera costruttiva. Ciò non toglie che determinata l'animazione di partenza, sia preferibile fare un bake dell'animazione sullo scheletro del personaggio (letteralmente 'cuocere' le animazioni, trasferirle nel rig di destinazione) e cancellare i vincoli con lo scheletro mocap in modo da alleggerire il peso del file.

La scelta di HumanIK per il rigging si è confermata come la migliore alternativa per poter lavorare con animazioni motion capture ed avere così la possibilità di effettuare il retargeting in maniera semi-automatica grazie a poche impostazioni come la definizione dei due rig. Un altro vantaggio è che il retargeting in questo stadio si può effettuare indipendentemente dalle dimensioni e proporzioni del modello, e in questo senso si è lavorato con la tranquillità che qualsiasi cambio nella topologia o nelle dimensioni di Nadya non avrebbe influito sul passaggio delle animazioni da uno scheletro ad un altro.

10.3.3 Retargeting del viso

Nel caso dell'animazione facciale, il 90% dell'animazione si è costruita in fase di retargeting con il software Faceware Retargeter. Importato il modello di Nadya si procede con la creazione del setup per il suo viso, quindi l'indicizzazione dei vari controllori responsabili delle deformazioni facciali. Tra i file prodotti dal programma si possono trovare i documenti in formato .xml e quindi facilmente leggibili, che vengono generati sia durante il setup del personaggio che durante la creazione di pose per il retargeting.

Per il setup del personaggio si possono distinguere le classi che compongono la sua configurazione: al componente *Head* non è stato assegnato alcun parametro perché la rotazione è affidata ai dati mocap della tuta inerziale, mentre si può notare dall'estratto del codice che il componente *mouth* ha diversi dati al suo interno. Il tag *ParameterFile* dichiara al programma quali dati fanno parte del gruppo corrispondente a quello di Analyzer (gruppo sopracciglia, occhi o bocca), in questo esempio stiamo analizzando il gruppo della bocca. Il tag *MovableAttribute* contiene al suo interno tutti i controllori che sono stati indicati come obiettivo per il retargeting, ai quali viene assegnato un ID riconoscibile per il programma e che quindi a volte può variare rispetto al nome dell'attributo originario. Il tipo di attributo è sempre lineare e il valore numerico che segue subito dopo rappresenta il momento a riposo dell'attributo, segnalato dal tag *defaultValue*.

Codice .xml per la configurazione del viso di Nadya, esempio:

```
<CharacterConfiguration>
  <Name>NadyaConfig</Name>
  <Component>
    <Name>Head</Name>
    <ParameterFile>Head_Params</ParameterFile>
  </Component>
  <Component>
    <Name>Mouth</Name>
    <ParameterFile>Mouth_Params</ParameterFile>
    <MovableAttribute>
      <Name>ctrlMouthCorner_L.translateX</Name>
      <ID>ctrlMouthCorner_L.translateX</ID>
      <AttributeType>Linear controls</AttributeType>
      <DefaultValue>
        <AttrValue>0.00109936</AttrValue>
      </DefaultValue>
    </MovableAttribute>
    <MovableAttribute>
      <Name>ctrlMouthCorner_L.translateY</Name>
      <ID>ctrlMouthCorner_L.translateY</ID>
      <AttributeType>Linear controls</AttributeType>
      <DefaultValue>
        <AttrValue>-0.00725</AttrValue>
      </DefaultValue>
    </MovableAttribute>
  </Component>
</CharacterConfiguration>
```

```

    </DefValue>
  </MovableAttribute>
<MovableAttribute>
  <Name>ctrlMouthCorner_R.translateX</Name>
  <ID>ctrlMouthCorner_R.translateX</ID>
  <AttributeType>Linear controls</AttributeType>
  <DefValue>
    <AttrValue>0.00193195</AttrValue>
  </DefValue>
</MovableAttribute>
[...]
```

Grazie al fatto che il rigging facciale è stato eseguito su ogni aggiornamento personaggio con gli stessi passaggi e utilizzando lo stesso script, il numero e il nome dei controller è sempre rimasto invariato, ed è così possibile caricare il profilo personaggio precedentemente creato avendo cura di aggiornare il valore dei *defaultAttribute* nel caso in cui durante le modifiche al modello 3D siano stati spostati alcuni vertici. Nella finestra a comparsa *Performance* si caricano il file .fwr contenente la parametrizzazione in uscita da Faceware Analyzer e la configurazione del personaggio, in modo che Retargeter sia in grado di maneggiare insieme all'utente i driver delle varie ossa del volto; si importa inoltre il video della performance per avere una reference durante il lavoro di retargeting.

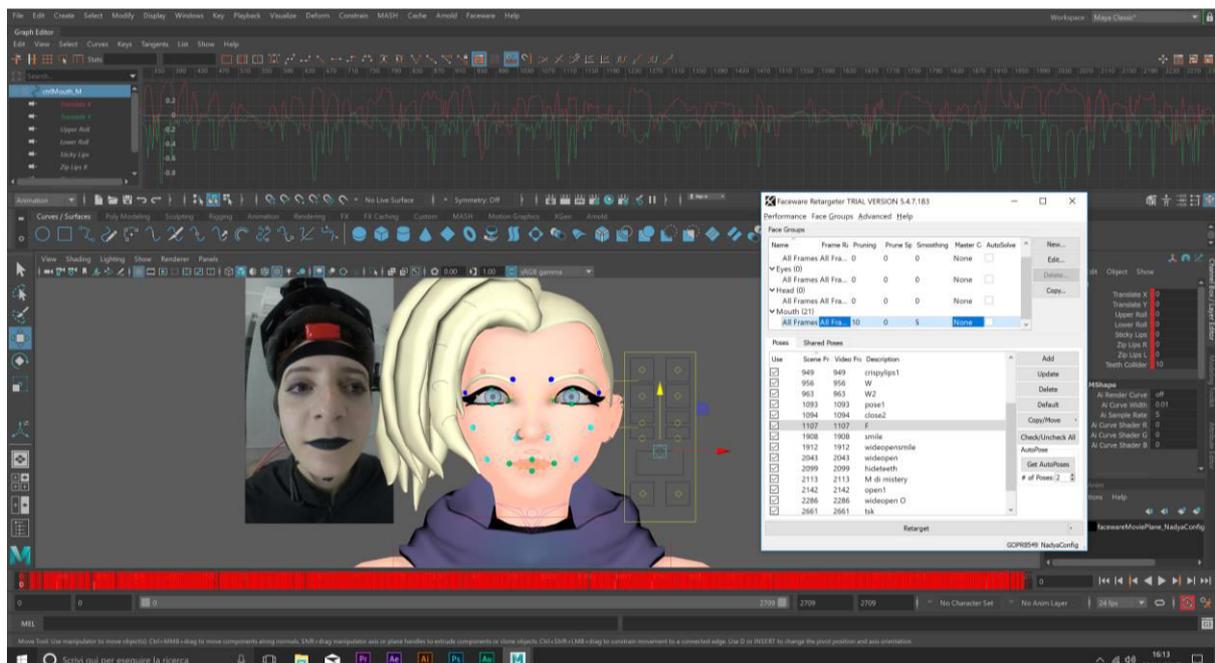


Figura 10.22: schermata complessiva di Maya e Faceware retargeter: la reference video permette di confrontare la somiglianza nella performance del personaggio 3D, in alto il grafico relativo al controller della bocca durante il monologo, nella finestra di Retargeter ci sono le pose salvate per il calcolo del retargeting.

In Retargeter, come abbiamo già menzionato, la prima sezione in alto mostra i gruppi di posa., che sono contenitori che mantengono le posizioni per ciascun rispettivo gruppo. Per esempio, le pose del sopracciglio verrebbero posizionate nel gruppo di posa del sopracciglio, la bocca si pone nel gruppo della bocca e così via. Questi gruppi si basano sui '*Rig Group*' che sono stati definiti nella configurazione del personaggio e ognuno di essi ha un numero variabile di controlli del rig. Si possono creare gruppi aggiuntivi ma tenendo a mente che prenderanno sempre le caratteristiche di quelli forniti di base. Questi gruppi sono in genere impostati su:

- Bocca
- Occhi
- Sopracciglia

Per effettuare il primo retargeting sono necessarie almeno due pose, quindi si procede a coppie di pose individuate nella parte del viso prescelta. Si chiede al programma di generare due pose, che verranno create analizzando il file .fwr e, grazie al video reference, si aggiusta l'espressione per farla combaciare il più possibile con quella dell'attore ripreso. Queste pose vengono create secondo i dati che Retargeter ha a disposizione dal suo programma compagno, Analyzer, ovvero le pose che inserirà nella libreria sono i migliori suggerimenti su dove impostare delle pose chiave. Molto spesso queste pose coincidono con espressioni del viso significative e distinte. E' possibile rinominare le pose individuate per mantenere un certo ordine di lettura e per distinguere le pose chiave come quelle per fonemi o espressioni particolari. Ogni volta che si aziona il retargeting, tutti i keyframe vengono cancellati e vengono calcolate le nuove coordinate per i controllori e rimappati in keyframe sulla timeline (come si vede in Figura 10.22 nella spessa linea rossa che rappresentano i keyframe salvati nel tempo). Si lavora localmente sugli attributi e si può effettuare il retargeting per il singolo attributo o per tutto il viso.

In questo frangente, l'errore generato dalle micro-vibrazione del caschetto si presentano chiaramente in scena, dando vita ad animazioni traballanti e a volte scattose, che a volte vengono accentuate da una densità esagerata di keyframe. Per poter ripulire questo tipo di rumore si può intervenire con tre modalità differenti, una manuale e le altre utilizzando funzionalità offerte da Faceware:

- 1) Si aggiungono pose chiave fino a quando il problema non scompare, ovvero si lavora sul dettaglio dei dati che il programma avrà a disposizione per il calcolo del retargeting;
- 2) *Pruning* - Il compito di questa funzione è rimuovere i keyframe dalla curva di animazione mantenendo la fedeltà del risultato. L'intenzione è quella di creare curve di animazione che siano facili da lavorare a mano per un animatore. Il software analizza le f-curve (curve di funzione) e rimuove in modo intelligente i keyframe che hanno il minore impatto sulla forma della curva.
- 3) *Smoothing* - Viene utilizzato per ridurre picchi o rumori taglienti dalle curve di animazione. L'algoritmo di smoothing funziona analizzando ogni keyframe e calcolando la media del suo valore in base ai valori delle chiavi che lo circondano. Da notare che rispetto al pruning, questa azione non comporta la cancellazione dei keyframe, ma solamente un calcolo della loro media per diminuire eventuali disturbi.

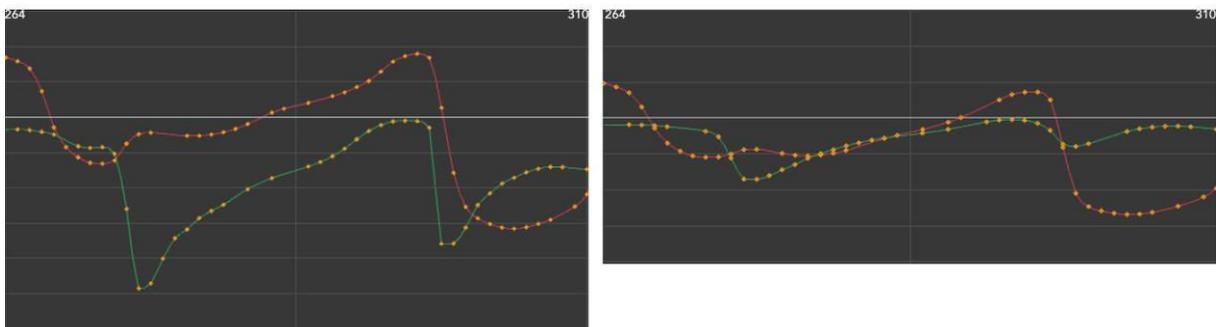


Figura 10.23: Esempio di pruning e smoothing per il driver globale della bocca. A sinistra la configurazione normale, a sinistra il grafico rielaborato con i valori 'pruning =15 e smoothing =10'.

I valori di pruning e smoothing hanno un range che va da 0-100 in percentuale, e bisogna trattarli con estrema attenzione per non rovinare il risultato qualitativo dell'animazione: è comodo poter suddividere dei range di keyframe da trattare con valori diversi di pruning e smoothing e se n'è fatto ampio uso per l'animazione di Nadya.

Come per Analyzer, è sconsigliato salvare più volte lo stesso tipo di espressione in valori temporali diversi e con un minimo di variazione nella posa perché il programma fa affidamento alle pose che gli sono state insegnate e quindi andrà a ricavare l'animazione totale partendo dalle sue pose chiave salvate.

Lavorando su l'animazione facciale di Nadya si sono compresi alcuni dei punti di forza di questa tesi, come il valore del motion capture per la pipeline di animazione di un

personaggio e la qualità offerta dall'utilizzo di questa tecnologia. Lo spunto di riflessione nasce pensando al tempo che si sarebbe impiegato a costruire manualmente l'animazione del lip-sync di Nadya rispetto al comporre un numero predefinito di pose sulla base di una performance attoriale dal vivo. Se a causa dei leggeri tremolii del caschetto i risultati del retargeting per le sopracciglia non si sono rivelate efficaci, per la bocca non si può dire lo stesso: il motion capture facciale è riuscito a restituire un'animazione delle labbra che è sorprendentemente fedele alla prestazione attoriale, e per un prodotto animato di questo tipo dove l'azione e l'emozione espressa si basa su un monologo in primo piano della protagonista, si è rivelato di fondamentale importanza per la riuscita di un prodotto altamente professionale.

Per concludere la trattazione del retargeting, nella produzione del trailer ci si è imposto il limite di pose chiave nell'ordine del centinaio per gruppo di posa, e si riassumono nella tabella 10.1 i valori utilizzati per l'elaborazione dell'animazione e il numero di pose salvate nella libreria:

| | Frame Range | Num Pose | Pruning | Pruning Space | Smoothing |
|-------------------------|-------------|----------|---------|---------------|-----------|
| Bocca | | 81 | | | |
| - <i>Starting point</i> | 0 - 214 | | 20 | 0 | 30 |
| - <i>AllFrames</i> | 215 - 2709 | | 15 | 0 | 10 |
| Sopracciglia | 0 - 2709 | 88 | 50 | 0 | 90 |
| Occhi | | 100 | | | |
| - <i>First part</i> | 0 - 343 | | 35 | 0 | 70 |
| - <i>Presentation</i> | 344 - 600 | | 50 | 0 | 90 |
| - <i>Second part</i> | 600 - 2709 | | 30 | 0 | 75 |

Tabella 10.1: Valori assegnati per il retargeting facciale

10.3.4 Animation Layers

Ottenuta la miglior performance possibile dal motion capture, è tempo di ottimizzare le animazioni tramite l'utilizzo dei layer di animazione. Uno dei motivi principali per cui i livelli di animazione sono stati integrati nella pipeline è quello di ripulire le compenetrazioni

di alcune parti di mesh che con il solo scheletro mocap non si potevano notare: le proporzioni del corpo di Maddalena e di Nadya presentano alcune differenze ed è quindi probabile che ci siano alcune discrepanze in fase di animazione, come le braccia che si intersecano con il petto o le gambe quando è seduta a terra. Un compito importante delegato all'animazione keyframe-base quindi è stato quello di ripulire tutti gli errori di animazione migrati dal motion capture, e tra queste correzioni emergono ulteriori pulizie sulla fluidità dei dati, soprattutto per quanto ha riguardato l'animazione facciale, degli occhi e delle sopracciglia.

I livelli di animazione consentono di creare e unire più livelli di animazione in una scena. È possibile creare livelli per organizzare nuove animazioni oppure per maneggiare fotogrammi chiave sopra l'animazione esistente senza sovrascrivere le curve originali. Questa organizzazione a livelli, tipica dei programmi di elaborazione di immagine, mantiene l'animazione nella scena con ogni livello contenente le curve di animazione per gli attributi che gli sono stati assegnati. I livelli appaiono impilati nell'Editor dei livelli di animazione e, in base alle diverse impostazioni, si fondono per creare l'animazione dei risultati che viene riprodotta nella scena. Le modifiche possono essere localizzate ad alcuni gruppi di giunti oppure possono prevedere delle modifiche all'animazione globale, rendendo facile esagerare, ridurre o modificare in altro modo l'animazione esistente, senza il rischio di perdere il lavoro originale. Quando l'animazione a livelli mostra una scena completa e corretta, si può scegliere di unire i livelli di animazione.

E' molto utile in una pipeline di produzione dove si richiedono mescolamenti tra varie animazioni generate dal motion capture, e per entrare nel dettaglio di alcune correzioni che di fronte a un grafico denso di data getterebbe nello sconforto anche l'animatore più veterano. I passaggi seguenti descrivono il flusso di lavoro che si è implementato per l'utilizzo di layer di animazione per modificare l'animazione mocap esistente:

- 1) L'animazione della tuta è impostata di default nel *Base Layer*;
- 2) Si selezionano i giunti di cui si desidera modificare l'animazione e si clicca sull'icona 'crea livello da selezione' (). Questo crea un livello di animazione per i giunti selezionati contenente gli attributi che si è deciso di lavorare: per impostazione predefinita, tutti gli attributi dell'oggetto vengono aggiunti al livello, ma tramite il *Channel Box* o l'editor delle relazioni è possibile aggiungere o rimuovere gli attributi di interesse;

- 3) Si manipolano i giunti e si impostano i keyframe sugli attributi a disposizione, i quali vengono impostati sul nuovo livello di animazione.
- 4) Si procede fino a quando non si ottiene l'animazione desiderata, dopodichè si può scegliere di fare un bake di tutti i livelli di animazione sul *Base Layer* oppure lasciarli separati in previsione di ulteriori modifiche ristrette.

Maya mette a disposizione degli utili strumenti per valutare i progressi nell'animazione, come il comando ghost (👻) che permette di visualizzare sia l'animazione originale che quella modificata con un effetto fantasma.

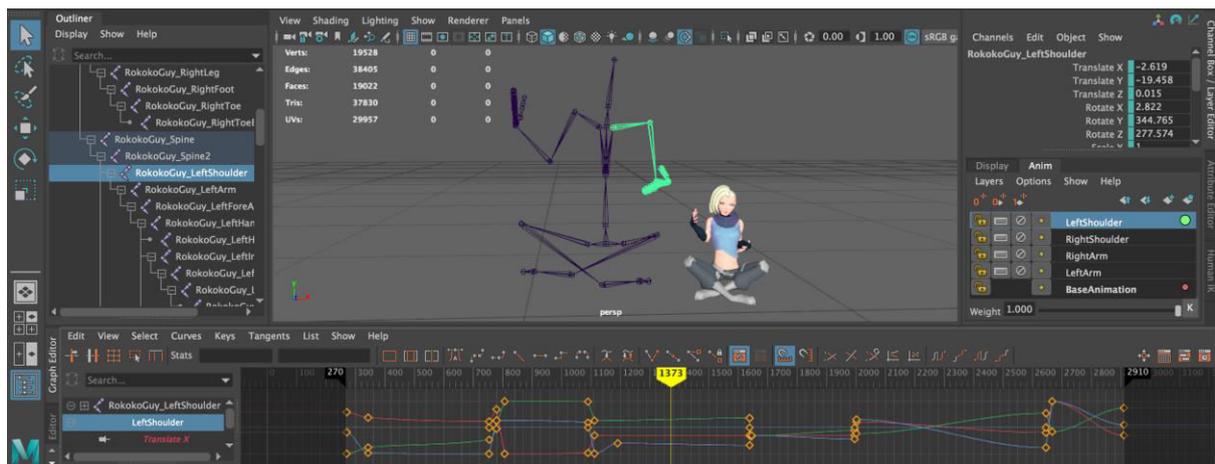


Figura 10.24: Animazione a livelli dello scheletro mocap, con il retargeting del modello che svolge la funzione di reference per le correzioni degli errori che dal solo scheletro non si possono vedere. Nel graph editor si può notare che i keyframe sono localizzati solamente nelle aree interessate dalle modifiche. La finestra a destra mostra una suddivisione per i giunti di interesse che si stanno modificando.

Un piccolo trucco per modificare alcuni errori di compenetrazione è quello di inserire un keyframe prima dell'inizio dell'errore nella posizione degli attributi come mostrata in scena, si ripete lo stesso procedimento per il keyframe subito dopo la fine dell'errore, e infine si inseriscono i keyframe necessari all'interno di quell'area per correggere la compenetrazione della mesh o qualsiasi altra modifica debba sovrascrivere l'animazione originale. Un'alternativa è l'assegnazione di un peso al layer di animazione ma questo non è utile allo scopo quando sono già presenti diversi keyframe sparsi nella timeline, perchè significherebbe aggiungere altri fotogrammi chiave per il valore del peso aggiungendo delle informazioni ridondanti o che si possono ottenere in maniera più immediata col metodo appena illustrato.

Per l'animazione di Nadya si è deciso di creare un livello di animazione per ogni giunto da correggere, per avere un controllo estremamente localizzato sugli attributi

strettamente necessari. In un paio di giornate di lavoro è stato possibile finalizzare l'animazione per il corpo e unire tutti i layer con un bake, che ha automaticamente risolto anche i residui problemi di jittering che si erano ereditati dai dati rovinati del mocap. I livelli per l'animazione facciale sono stati fondamentali in questo senso, perchè il tremolio provocato dall'instabilità della camera si sono ripercossi in una animazione rumorosa per gli occhi e per le sopracciglia. Nel secondo caso è bastato valutare dove le sopracciglia si inarcavano, o si increspavano, e si sono aggiunti i keyframe laddove era necessario esagerare l'animazione. Per quanto riguarda gli occhi, il problema da risolvere stava nello sguardo di Maddalena, che puntava alla camera oggetto di registrazione del trailer, ma molto spesso la direzione dello sguardo non coincideva con la posizione della GoPro sul caschetto fisico, e quindi è stato necessario correggere con un layer di animazione la direzione dello sguardo perché puntasse sempre verso il video a cui la protagonista si sta rivolgendo.

Con l'applicazione di animazione keyframe-based all'animazione mocap è stato possibile aspirare a un risultato qualitativo importante, difficilmente raggiungibile con una buona conoscenza dei principi dell'animazione tradizionale mai applicati a un progetto di questa portata. Il motion capture svolge un ruolo chiave in un ambiente indipendente emergente, dove le competenze si stanno creando dal basso ma si deve comunque rientrare nelle scadenze stabilite dalla produzione.

10.4 Rendering del prodotto finale

L'aspetto grafico finale del teaser è ancora in intensa fase di sviluppo ed è portato avanti dallo sforzo congiunto del reparto tecnico, ovvero dalla sottoscritta e dal mio collega Michele Cannata, sotto la supervisione del nostro relatore.

Attualmente stiamo studiando i motori di rendering alla ricerca di quello che ospiterà la nostra scena. Uno dei vincoli che ci è stato richiesto di rispettare è l'utilizzo di un motore che lavori in GPU invece che con la sola CPU, per velocizzare le prestazioni del rendering ma questo comporta alcuni problemi tecnici in quanto la grande maggioranza dei motori in GPU non supportano le schede grafiche Radeon, sui cui il computer personale è assemblato, rendendo impossibile di fatto aiutare nella fase sperimentale e dei test di rendering. A questo si aggiunge lo stile grafico che si è discusso all'inizio del capitolo, uno stile cartoon ma arricchito da texture che simulano delle pennellate per caratterizzare maggiormente lo stile e renderlo unico. Un altro aspetto da tenere a mente è la tipologia dell'illuminazione: si ricerca

un taglio delle luci che sia netto, con la possibile aggiunta di una rim light per creare un contorno di luce attorno al personaggio.

In considerazione di tutti queste note di produzione, ci siamo impegnati nel ricercare il metodo più efficiente per i calcolatori a nostra disposizione per il rendering e abbiamo iniziato a testare diversi modelli di shading, cioè di materiali, per comporre la scena. Uno dei primi risultati che ha suscitato un certo interesse ha visto l'utilizzo di alcuni shader toon impostati sull'angolo della luce, o sull'angolo delle facce ovvero l'illuminazione tende a variare dal punto di vista della camera.

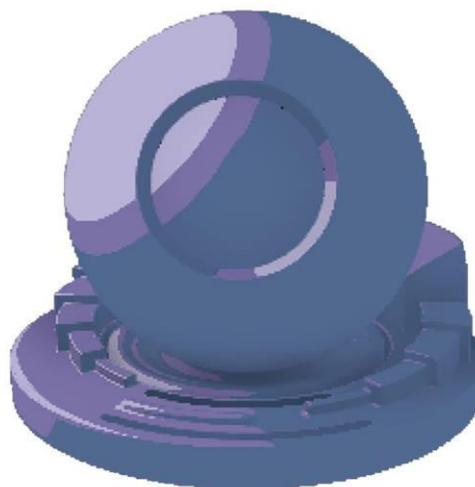
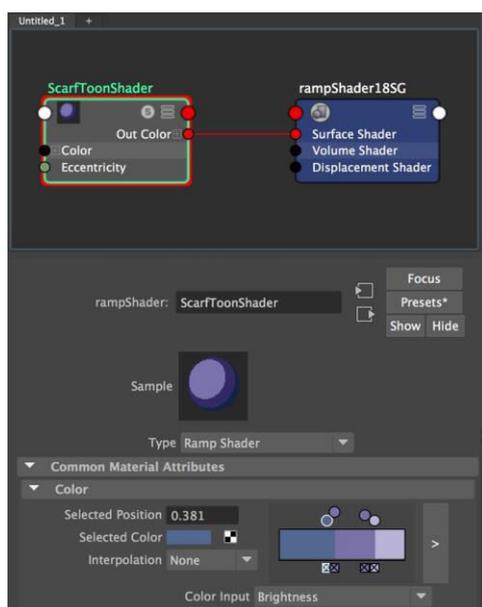


Figura 10.25: Toon shader di prova per l'oggetto sciarpa e per la base colore di Alep

I ramp shader in particolare, permettono di simulare lo stile cartoon con ampio margine di impostazione, giacché è possibile aggiungere toni alla rampa e aumentare così il livello di sfumature con cui il materiale colorerà l'oggetto. Esplorando questa via abbiamo provato ad aggiungere al ramp shader la texture creata da Michele per il viso, dove inizia a intravedersi l'effetto pennellata, e abbiamo ottenuto un risultato come in Figura 10.26: la pennellata perde di valore perché si intravede appena nel rendering del modello, però dona al ramp shader tutta una serie di sfumature che sono oggettivamente apprezzabili. Da un lato si è proseguito per questa via di ricerca, perché l'utilizzo del playblast di Maya per il rendering si rivela essere uno strumento veloce di esportazione di livelli di rendering dell'animazione che poi si possono assemblare in After Effects e maneggiare per il compositing. Dall'altro lato si sono

ricercati metodi alternativi che rispondessero all'esigenza di un motore di rendering performante e veloce, possibile che lavorasse con la GPU.

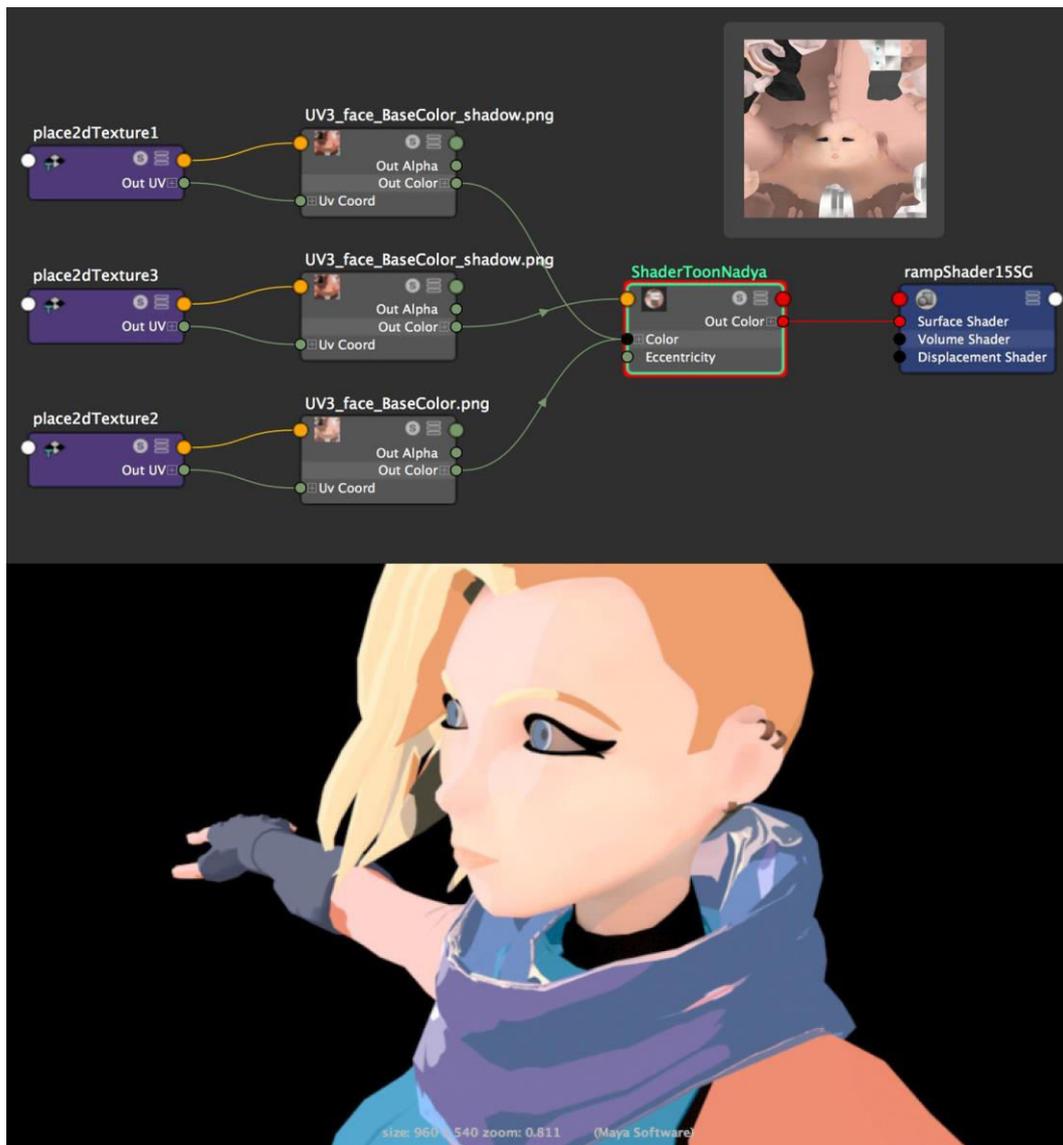


Figura 10.26: Toon shader con texture per il viso di Nadya, e rendering in Maya Software

Si sono iniziati a delineare i livelli di rendering che sono necessari da esportare per poter ottenere un effetto stilistico come richiesto, e i livelli in questione sono i seguenti:

- Base Color - il materiale di base contenente la texture del personaggio con le pennellate accentuate e ancora più grezze;
- AO - ambient occlusion, una tecnica di shading e rendering usata per calcolare come l'esposizione di ogni punto di una scena rispetto all'illuminazione ambientale, ovvero

è in grado di calcolare localmente il grado di attenuazione dovuto a fattori di occlusione;

- *Ramp Shader RGB light angle* - si vuole utilizzare una rampa basata sull'angolo della luce configurata con i canali R, G e B, che una volta importata su After Effects sarà la base di utilizzo per diversi filtri di curve, saturazione e illuminazione localizzati sul singolo canale.
- Rim light - un livello per aggiungere un contorno di luce al personaggio, dandogli volume e aumentandone il realismo.

La composizione di questi livelli porterà alla realizzazione di una scena che rappresenterà l'avvicinamento al target grafico che ci si è immaginati e imposti, creando nel frattempo una pipeline di rendering dedicata, e quindi unica.

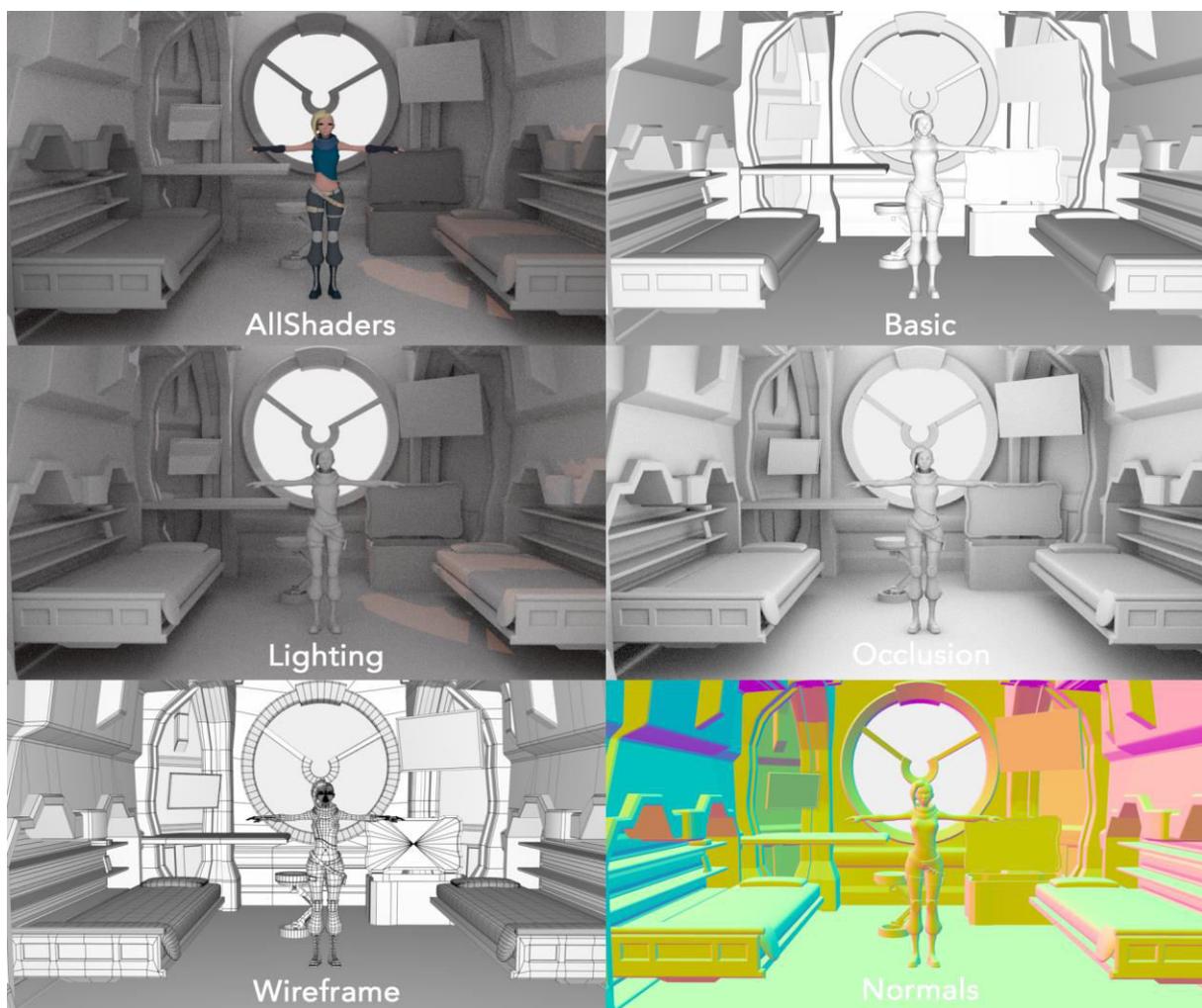


Figura 10.27: Debug render pass di Arnold

Ad oggi, metà novembre e a un mese appena dalla presentazione della prima versione del trailer, si stanno ancora effettuato svariate prove di materiali e rendering, e i primi test di esportazione completi di ambientazione sono molto promettenti.

Si sta studiando il nuove motore integrato in Maya, Arnold, che sarebbe in grado di offrire lo stesso tipo di trattamento dello shader *toon ramp* ma con opzioni specifiche. L'enorme vantaggio offerto da questo motore di rendering, nonostante non offra al momento il supporto alla GPU, è che nella finestra di render view è possibile pre-visualizzare di shading utili alla composizione della scena finale, come l'AO (ambient occlusion), la normal map e il wireframe. Offre quindi uno strumento di visualizzazione temporaneo e veloce, prima di passare alla creazione vera e propria di livelli di rendering che applichino effettivamente lo shading desiderato.



Figura 10.28: Test di rendering in playblast con ambientazione, luci direzionali e toon shader applicati a tutti i modelli (cliccare sull'immagine per andare al video)

In definitiva, quello del rendering è un argomento di studio ancora aperto, ma lo scorgersi dei primi esiti positivi anche nella resa visiva, oltre che nell'animazione, aumentano esponenzialmente il valore del lavoro svolto durante questi mesi di lavoro e ricerca in tanti ambiti diversi, e tutto il team di produzione non vede l'ora di mostrare al pubblico i risultati di questo ambizioso progetto.

11. Conclusioni

Sviluppare un prodotto di animazione è un percorso davvero molto lungo, soprattutto nel contesto di un'azienda nata da poco le cui competenze primarie dei collaboratori non seguono principalmente quel determinato reparto tecnico e creativo. All'interno di questo progetto di tesi c'è stata un'immensa e inestimabile possibilità di sperimentare autonomamente nella ricerca e nella metodologia che ha portato alla costruzione della pipeline che si è descritta. Non si è trattato solo di uno sviluppo creativo ma, con la necessità ad esempio di trovare una soluzione per registrare il salto in motion capture si è aperta la via della ricerca pura in ambito tecnologico e matematico, alla base della conoscenza di un ingegnere. In questi nove mesi si è studiata approfonditamente la tecnologia del Motion Capture, imparando con un approccio sperimentale quali siano i suoi limiti d'utilizzo e stimolando il pensiero su come superare questi limiti. Il fatto di lavorare in un contesto indipendente è stata un'ulteriore spinta per trovare delle soluzioni semplici ma robuste e con un buon grado di difficoltà tecnica, sostenibili economicamente e riutilizzabili per diversi ambiti di progetto. Lo studio sul motion capture ha costituito uno studio approfondito e rigorosamente autonomo della materia, e le conoscenze scaturite sono state condivise all'interno dell'ambiente di lavoro come un valore aggiunto a competenze già consolidate. La volontà di sperimentare software e pacchetti grafici completamente nuovi nel nostro contesto universitario ha rappresentato un ulteriore livello di sfida da superare mentre si apprendevano nozioni tecniche e buone pratiche d'utilizzo, ma con l'obiettivo di introdurre all'interno dell'Ateneo gli stessi programmi professionali che nell'industria di alto livello vengono comunemente applicati, e quindi richiesti in fase di colloquio. In questo senso, si sono sviluppate delle conoscenze, e delle competenze, altamente competitive per il mercato dell'animazione, indipendente e non. Allo stesso tempo c'è stata la possibilità di creare una rete di connessione con Rokoko e il loro team di sviluppo, e anche solo l'idea del viaggio-training a Copenaghen è stato un risvolto meravigliosamente positivo che si è sviluppato nel corso delle ricerche, e che inizialmente non era stato neanche contemplato come possibilità. La possibilità di espandere gli orizzonti al di fuori delle mura accademiche e di creare una rete di collaborazione con esperti del settore a livello internazionale, ha sicuramente aggiunto valore all'argomento della tesi che si è trattata. Con l'inizio della produzione del teaser trailer, ci si è assunti la responsabilità di coordinare una piccola squadra di produzione e l'ostacolo più difficile da superare durante questo tipo di lavoro è stato quello di gestire momenti di crisi, dietrofront

nella pipeline e soprattutto la comunicazione all'interno del gruppo. C'è ancora ampio margine di miglioramento sotto questo aspetto, una lezione fondamentale che si è appresa da questa esperienza è il saper gestire sapientemente il tempo che si ha a disposizione. Col senno di poi, alcune decisioni si sarebbero potute prendere in maniera più rilassata e il risultato si sarebbe tradotto in un lavoro più sereno e meno ripetitivo, ed è un punto sul quale si intende continuare a lavorare nel futuro prossimo.

Perché il progetto *Reverie Dawnfall* non è solamente il fine di questa ricerca ma anzi, lo scopo del teaser trailer è stato quello di iniziare a sperimentare diverse tecniche di modellazione, animazione e rendering per arrivare alla creazione di una vera e propria pipeline di produzione animata. Un esperimento che è stato affrontato con spirito creativo, flessibile e con estremo entusiasmo, elemento imprescindibile per non cedere allo sconforto di fronte all'accavallarsi di scadenze e intoppi tecnici.

Quello che la mia tesi ha voluto dimostrare è come è possibile integrare il motion capture nell'animazione e velocizzare così il percorso produttivo mantenendo un elevatissimo grado di realismo, e migliorare sul grado qualitativo per poter raggiungere livelli altamente professionali, nonché stimolanti. Sul lato strettamente personale, ho avuto modo di mettere in gioco le conoscenze teoriche apprese durante i due anni di questo percorso magistrale in un campo che si intende trasformare nella mia professione: all'interno dello studio Robin sarò in grado di svolgere un ruolo di sempre maggiore responsabilità nel reparto di animazione e del motion capture, offrendo tutte quelle competenze guadagnate da questa esperienza a clienti e collaboratori. Con Rokoko si è instaurato un rapporto di collaborazione continuativa, come tester per la loro tecnologia, e per quelle future. E ancora, da questa esperienza ho avuto modo di apprendere davvero tantissimo a livello tecnico, interpersonale e professionale. L'unico rimpianto è quello di non aver intrapreso molto prima questo tipo di percorso, perché con quello che si sta continuando ad imparare ogni giorno sul motion capture e sul mondo dell'animazione si arriverà a creare veramente delle *'robe incredibili'*.

Appendice A: Filtro di Eulero e Quaternioni per la prevenzione del Gimbal Lock

Nella grafica computerizzata, il filtro di Eulero è un filtro destinato a impedire il blocco cardanico e le discontinuità correlate nei set di dati di animazione in cui la rotazione è espressa in termini di angoli di Eulero. Il blocco del giunto cardanico (o *gimbal lock*) è la perdita di un grado di libertà in un meccanismo tridimensionale a tre giroscopi, che si verifica quando gli assi di due dei tre assetti cardanici si inseriscono in una configurazione parallela, *bloccando* il sistema in una rotazione nello spazio degenerato a due dimensioni.

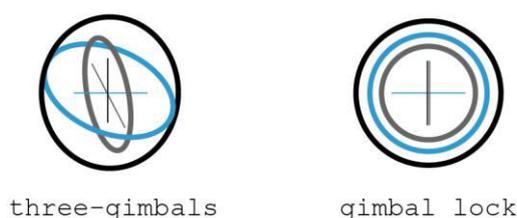


Figura A.1: Esempio di gimbal lock

La parola lock è fuorviante poiché nessuna sospensione cardanica è realmente trattenuta: tutti e tre i giroscopi possono ancora ruotare liberamente attorno ai rispettivi assi di sospensione tuttavia, a causa dell'orientamento parallelo di due degli assi dei giunti cardanici, non è disponibile una sospensione cardanica per accogliere la rotazione lungo un asse, creando quindi una discontinuità nei dati come si è potuta notare durante la performance analysis della tuta Smartsuit. Queste discontinuità sono perciò causate dall'esistenza di mappature multi-uno tra la parametrizzazione dell'angolo di Eulero dell'insieme di rotazioni 3D. Ciò consente al set di dati di scegliere tra diverse combinazioni di angolo di Eulero che corrispondono a una singola rotazione 3D che, pur rimanendo continua nello spazio di rotazione, sono discontinue nello spazio dei parametri dell'angolo di Eulero.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 1 & \frac{\sin(\phi) \sin(\theta)}{\cos(\theta)} & \frac{\cos(\phi) \sin(\theta)}{\cos(\theta)} \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \frac{\sin(\phi)}{\cos(\theta)} & \frac{\cos(\phi)}{\cos(\theta)} \end{bmatrix} \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix}$$

Formola (1): Descrizione matematica per il blocco cardanico facendo uso degli angoli di Eulero. Le singolarità si presentano quando i denominatori sono nulli, ovvero $\theta = \pm \frac{\pi}{2}$.

Il filtro Eulero è disponibile in numerosi pacchetti di animazione 3D, e funziona scegliendo una base campione per campione tra le possibili rappresentazioni dell'angolo di Eulero di ciascuna rotazione 3D nel set di dati in modo tale da preservare la continuità delle serie temporali dell'angolo di Eulero, senza modificare le effettive rotazioni 3D.

Un altro modo per risolvere il blocco cardanico è quello di usare un diverso sistema di rotazione, e la soluzione più comune in questo frangente è quella dei quaternioni, un'estensione dei numeri complessi che introducono un'ulteriore coordinata spaziale. I quaternioni possono rappresentare angoli di rotazione e movimento senza i problemi del blocco cardanico. Questo è possibile quando i quaternioni di unità sono mappati sull'ipersfera in uno spazio quadridimensionale. Il quaternione di unità è rappresentato dalle seguenti formule:

$$q = w + xi^2 + yj^2 + zk^2 \quad (1)$$

$$1 = w^2 + x^2 + y^2 + z^2 \quad (2)$$

In questo caso, l'angolo di rotazione $w = \cos(\Theta / 2)$ e il vettore (x, y, z) rappresentano il punto (w, x, y, z) . Quindi l'equazione per l'interpolazione di quaternioni è:

$$Q(q_0, q_1, t) = (q_1) (q_0^{-1})^t (q_0) \quad (3)$$

dove t è tra 0 e 1, muovendosi lungo il percorso lineare del grande cerchio associato.

Bibliografia

Bibliografia classica

- [1] Kitagawa, M., & Windsor, B. (2008). “MoCap for artists: workflow and techniques for motion capture”. Elsevier/Focal Press. ISBN: 978-0-240-81000-3
- [2] Mecheri, Hakim & Robert-Lachaine, Xavier & Larue, Christian & Plamondon, Andre. (2016). “Offset correction when comparing 3D joint angles from two different motion capture systems”. In Conference: *Digital Human Modeling*, At Montréal, Québec, Canada
- [3] Orvalho, V. C., & Susin, A. (2007). “Fast and reusable facial rigging and animation”. *ACM SIGGRAPH 2007 Sketches on - SIGGRAPH '07*, <http://doi.org/10.1145/1278780.1278856>
- [4] Parent, R. (2012). “Computer animation: algorithms and techniques”. Morgan Kaufmann Publishers. ISBN-10: 0124158420, ISBN-13: 978-0124158429
- [5] Robert-Lachaine, X., Mecheri, H., Larue, C., & Plamondon, A. (2017). “Effect of local magnetic field disturbances on inertial measurement units accuracy”. In: *Applied Ergonomics*, 63, 123–132. <http://doi.org/10.1016/j.apergo.2017.04.011>
- [6] Roetenberg, D 2006, “Inertial and magnetic sensing of human motion”, University of Twente, Enschede.
- [7] Roy, K. (2014). “How to cheat in Maya 2014: Tools and Techniques for Character Animation”. Focal Press. ISBN-10: 0415826594, ISBN-13: 978-0415826594
- [8] Szczesna, A., Skurowski, P., Lach, E., Pruszowski, P., Pęszor, D., Paszkuta, M., Wojciechowski, K. (2017). “Inertial Motion Capture Costume Design Study”. In: *Sensors*, 17(3), 612. <http://doi.org/10.3390/s17030612>

Sitografia

A Brief History of Motion Capture for Computer Character Animation, http://www.siggraph.org/education/materials/HyperGraph/animation/character_animation/motion_capture/history1.htm

All Time Worldwide Box Office, <http://www.the-numbers.com/box-office->

[records/worldwide/all-movies/cumulative/all-time](#)

Benedict Cumberbatch - The hobbit - The best scene - Smaug-. (2015, August 11),
<https://youtu.be/sXN9IHrnVVU>

Bernstein, J. (2003, February 1). “An Overview of MEMS Inertial Sensing Technology”,
<http://www.sensorsmag.com/components/overview-mems-inertial-sensing-technology>

Dent, S. (2016, July 14). “What you need to know about 3D motion capture”,
<https://www.engadget.com/2014/07/14/motion-capture-explainer/>

Faceware Technologies, <http://facewaretech.com>

Hoag, D. (1963, April), “Apollo Guidance and Navigation Considerations of Apollo IMU Gimbal Lock”, <https://www.hq.nasa.gov/alsj/e-1344.htm>

Home | Autodesk Knowledge Network, <https://knowledge.autodesk.com/>

Jordan Zakarin on June 21, 2016. How 'Final Fantasy' Changed Hollywood,
<https://www.inverse.com/article/17234-the-final-fantasy-film-bombed-but-its-motion-capture-revolutionized-hollywood>

Levinson, M., “The Quaternions”. http://ffden-2.phys.uaf.edu/webproj/212_spring_2014/Michael_Levison/Michael%20Levison/rotation/index.html

Maga Animation, <http://www.maga-animation.com/work/ubisoft-mario-rabbids-kingdom-battle>

Nickromano. (2014, July 14). “A Visual History of Motion-Capture Performances on Film”,
<http://screencrush.com/motion-capture-movies/>

Rokoko, <http://www.rokoko.com/>

Salto - Captures your every move. VR, AR, film, VFX, gaming,
<https://www.kickstarter.com/projects/2132274632/salto-captures-your-every-move-vr-ar-film-vfx-gami>

Thomas, I. (2016, March 6). “10 Best Motion Capture Performances Ever”,
<https://screenrant.com/best-motion-capture-performances-movies/>

Ulbrich, E., “How Benjamin Button got his face,”
https://www.ted.com/talks/ed_ulbrich_shows_how_benjamin_button_got_his_face#t-

590813

Waldo - The Jim Henson Hour - The Jim Henson Company. (2011, March 31),
<https://youtu.be/dP6TUB7KQc4>